NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Md. 20034

A COMPUTER PROGRAM THAT USES INTERACTIVE GRAPHICS TO SOLVE
INVISCID TRANSONIC FLOWS OVER LIFTING AIRFOILS

by

Douglas A. Lien

and

Tsze C. Tai

FOR OFFICIAL USE ONLY

AVIATION AND SURFACE EFFECTS DEPARTMENT
RESEARCH AND DEVELOPMENT REPORT

September 1973 Report 4252

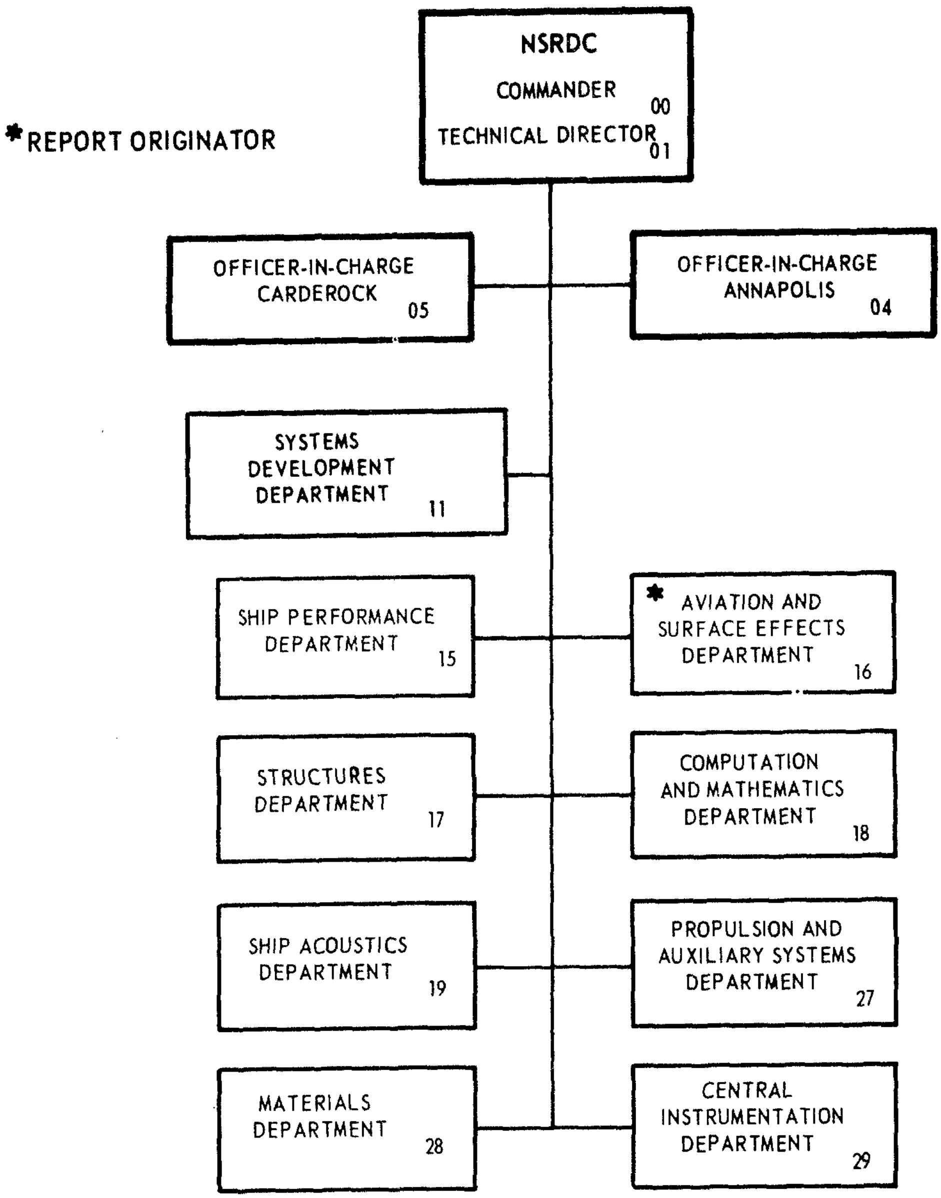
A COMPUTER PROGRAM THAT USES INTERACTIVE GRAPHICS TO SOLVE
INVISCID TRANSONIC FLOWS OVER LIFTING AIRFOILS

The Naval Ship Research and Development Center is a U. S. Navy center for laboratory effort directed at achieving improved sea and air vehicles. It was formed in March 1967 by merging the David Taylor Model Basin at Carderock, Maryland with the Marine Engineering Laboratory at Annapolis, Maryland.

Naval Ship Research and Development Center

Bethesda, Md. 20034

MAJOR NSRDC ORGANIZATIONAL COMPONENTS



DEPARTMENT OF THE NAVY

NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Maryland 20034

A COMPUTER PROGRAM THAT USES INTERACTIVE GRAPHICS TO SOLVE INVISCID TRANSONIC FLOWS OVER LIFTING AIRFOILS

by

Douglas A. Lien

and

Tsze C. Tai

FOR OFFICIAL USE ONLY

September 1973

Report 4252

Aero Report 1194

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

A computer program that utilizes the method of integral relations has been developed at the Naval Ship Research and Development Center for use in determining the inviscid transonic flows past lifting airfoils. It allows for a change of entropy across the shock wave and accounts for the presence of an oblique or normal shock at the shock foot. Since many iterations of the trial and error type are required to obtain the converged flow solution, the program has been adapted for use on the interactive graphic systems of the CDC 6700 computer. This minimizes the man-machine interaction time involved with such iterations. It has been applied to several airfoil cases with supercritical flow on the upper surface and subcritical flow on the lower surface and takes about 5 to 10 min of computer time per case. The theoretical basis for this program has previously been reported. This report documents the computer program which is written in the language of FORTRAN Extended Version 3.0.

ADMINISTRATIVE INFORMATION

This work was sponsored by the Naval Air Systems Command (NAVAIR-320) and funded under NAVAIR Task R230.201, Work Unit 1-1670-277.

INTRODUCTION

Application of the method of integral relations to solve transonic flow problems has already been developed and the method used in several flow solutions.[1,2] The present report documents the subroutines used in

1. Tai, T. C., "Application of the Method of Integral Relations to Transonic Airfoil Problems: Part I — Inviscid Supercritical Flow over Symmetrical Airfoil at Zero Angle of Attack," NSRDC Report 3424 (Sep 1970); also presented as Paper 71-98, AIAA 9th Aerospace Sciences Meeting, New York, N.Y. (Jan 1971).

2. Tai, T. C., "Application of the Method of Integral Relations to Transonic Airfoil Problems: Part II — Inviscid Supercritical Flow About Lifting Airfoils with Embedded Shock Wave," NSRDC Report 3424 (Jul 1972); also presented as Paper 73-658, AIAA 6th Fluid and Plasma Dynamics Conference, Palm Springs, California (Jul 1973).

computing transonic flows and illustrates their use with two examples: transonic flow past an NACA 0015 airfoil at $\alpha = 4.0$ deg and transonic flow past an advanced airfoil at $\alpha = 1.5$ deg.

The solution procedure consists of ten well-defined steps in accordance with necessary iteration processes. The completion of each step must satisfy certain flow conditions before the next step is undertaken. Actually, there are only three major iteration processes which form the bulk of the flow integration, and each process can be computed rapidly and efficiently. The main drawback to this method is that each step must be computed separately and that the output of one step is needed before the next step can proceed. This can be a time-consuming process if done by conventional means.

The use of interactive graphics greatly reduces man-machine interaction time. The input parameters and program execution can be modified by using a system of light registers and light buttons displayed on the CDC 274 graphics console screen. In order to simplify the solution process, only subcritical flow on the lower surface and supercritical flow on the upper surface will be allowed.

The primary inputs to the program are the airfoil coordinates (a maximum of 40 data points) and 32 extraneous and physical flow parameters. During execution of the interactive graphics program, 18 of these flow parameters may be changed, but ordinarily only one or two are used to iterate on a particular flow solution to satisfy a particular flow condition. The remainder of the flow solution parameters may be properly determined subject to the necessary constraints.

The importance of a well-defined airfoil shape cannot be stressed too strongly. This highly sensitive technique requires great accuracy in first and second derivative information from the airfoil surface. The spline function is one highly recommended method for representing airfoil surfaces. It can attain very accurate first and second derivatives from the airfoil surface if certain constraints are chosen judiciously. The method is explained in detail in Appendix A.

However, one fact should be borne in mind before attempting to use this program to solve transonic flows; it is not a "black box" computer program which generates output for a given set of input data. It requires special attention during execution to ensure that certain flow requirements are met. If the calculated flow is unsatisfactory, one of the input parameters should be changed to yield a satisfactory result. Luckily, it can be seen from inspection whether the value of a parameter is too large or too small, and input changes can be made accordingly.

DESCRIPTION OF COMPUTER PROGRAM

Application of the method of integral relations for transonic flow problems involves three major flow solutions:

1. Upstream solution
2. Airfoil solution
3. Downstream solution

These solutions must be computed sequentially, that is, the upstream solution must be computed before proceeding to the airfoil solution, and the airfoil solution must be computed before proceeding to the downstream solution. These steps are shown in Figure 1. The order of operations within the airfoil solution is immaterial; either the upper surface flow or the lower surface flow can be computed first.

The input to the program is only an approximation to the correct input which would yield a satisfactory solution. During the course of the solution, the input is modified to satisfy certain flow conditions. For instance, in the case of supercritical flow in the airfoil solution, the initial condition parameter CYD is changed until calculations show that the velocity gradient is continuous through the sonic point. Other inputs are modified in reply to the questions shown in Figure 1. When all the flow conditions are met satisfactorily and the solution is complete, the calculated pressure distribution is the serendipitous result of the solution process.

ORDER OF OPERATIONS

The flow chart of Figure 2 gives a more detailed analysis of the order of operations of the more important subroutines. A list of these subroutines and their function is given below.

UPSTRM = performs upstream flow integration

STAGNA = calculates stagnation streamline geometry and cross velocity gradient for given stagnation point XS

UPRCRIT = calculates Mach number conditions along initial portion of upper surface

LWRCRIT = calculates Mach number conditions along initial portion of lower surface

UPRINIT = calculates initial conditions on upper surface for a selected initial point and CYD

LWRINIT = calculates initial conditions on lower surface for a given initial point and CYD

SUBCRT1 = performs subcritical flow integration on initial portion of either upper or lower surface

SUBCRT2 = performs subcritical flow integration on either upper or lower surface

SPRCRT1 = performs supercritical flow integration on initial portion of upper surface

SPRCRT2 = performs supercritical flow integration on upper surface

DWNSTRM = performs downstream flow integration

AKUTTA = provides outputs of calculated upper and lower surface pressure distributions

Subroutines UPSTRM and STAGNA correspond to the upstream solution, and subroutines DWNSTRM and AKUTTA correspond to the downstream solution of Figure 1. The rest of the subroutines correspond to the airfoil solution. The details of the subroutines are given in Appendix B, and a flow chart of each subroutine is given in Appendix C.

Eight of the more important decision points are numbered in Figure 2. The dotted lines indicate the parameter changes needed for satisfactory results. Each decision point requires some attention, either a

modification of the input parameters or a decision on which course to follow in the computation.

An overview of the solution process which consists of various steps is given in Figure 3. Since the flow solution on the upper surface is much more interesting than that on the lower surface, only upper surface flow is discussed here in detail.

The numbered stars around the airfoil correspond to certain subroutines in Figure 2:

1	UPSTRM
2	STAGNA
3	UPRINIT-SPRCRT1
4	SPRCRT2
5	DWNSTRM
6	AKUTTA

The output for a particular subroutine is on either side of the corresponding number in the figure. The trial solution to the left could be improved on; the arrow indicating the parameter changes needed to improve the solution, and the corrected or acceptable solution to the right represents a completed step. Once this is completed, the program begins executing the next step.

The output from the first step shows a plot of Y versus Mach number. This velocity profile is taken from the final integration station of subroutine UPSTRM. The number of strips used to integrate the flow solution in the trial solution proved inadequate, and more strips were added to yield the corrected solution.

The second step is concerned with the selection of a stagnation point. The stagnation point for the trial solution was chosen at the nose of the airfoil; this yielded unrealistic stagnation streamline geometry for a lifting airfoil. A more satisfactory location of the stagnation point is given in the corrected solution. The selection of the stagnation point is most critical to later calculations.

The first major iteration process is given in the third step. In the trial solution the initial condition parameter CYD, which depends on assumed velocity profile shape ahead of the airfoil, did not yield velocity

gradients which were continuous through the sonic point. In the case of CYD = 1.0, the flow accelerated too rapidly before the sonic point, and in the case of CYD = 1.010, the flow decelerated before the sonic point. CYD = 1.005 for the corrected solution, and the velocity gradients were continuous through the sonic point. This step calculates the flow on the initial portion on the upper surface. The fourth step calculates the remainder of the flow.

In the fourth step, the only requirement for a satisfactory solution is the selection of a shock location which allows the flow calculations to proceed to the trailing edge. The shock location must be chosen so that the flow behind it remains subcritical throughout to the trailing edge. The exact shock location is determined by satisfying the downstream flow condition as outlined in the fifth step. The two initial guesses in the trial solution show cases of flow which become supercritical again after the shock wave. The corrected solution indicates where a case flow remains subcritical behind the shock location.

The third and fourth steps constitute the airfoil solution on the upper surface. For the lower surface of the airfoil, a solution is sought which allows flow integration to proceed to the trailing edge. Once the airfoil solutions for the upper and lower surfaces have been obtained, the downstream solution may be calculated.

The fifth step is concerned with flow calculations downstream from the airfoil. In the trial solutions the pressures diverged from the free-stream pressures quite rapidly. Thus it was necessary to return to the fourth step and select a new shock location which would yield down-stream pressures bracketing the free-stream values. It can be seen in the corrected solution that the final shock location was between 0.50 and 0.51; the pressure was slightly greater that free-stream pressure for one value and slightly less for the other.

The final step of the solution process is to check the calculated pressure distributions on the upper and lower surfaces of the airfoil. If the pressures at the trailing edge do not match on the upper and lower surfaces, the Kutta condition is not met, and program control should be transferred to the second step for the selection of a new stagnation point. If the stagnation point is judiciously chosen, the pressure distribution in the corrected solution should appear.

ILLUSTRATIVE EXAMPLES

A description of the order of operations of this computer program is best presented by illustrating its application to a particular airfoil.

NACA 0015 Airfoil

The NACA 0015 airfoil at an angle of attack of 4 deg and a free-stream Mach number of 0.729 are used here for purposes of illustration. Experimental results have shown that at these flow conditions, the flow is supercritical on the upper surface and subcritical on the lower surface.[3]

The five input flow parameters of greatest importance to the solution process are:

- DVOOI $(dV_o/ds)_o$, the estimated cross velocity gradient at the stagnation point,
- XS, the X-coordinate of the stagnation point,
- CYDL, the initial condition parameter for the lower surface flow,
- CYDU, the initial condition parameter for the upper surface flow, and
- SL, the location of the shock foot for the upper surface flow.

Decision point 1 comes after subroutine STAGNA, the calculation of stagnation streamline geometry and the cross velocity gradient at the selected stagnation point. It is important to select a stagnation point

3. Graham, D. J. et al., "A Systematic Investigation of Pressure Distribution at High Speeds over Five Representative NACA Low-Drag and Conventional Airfoil Sections," NACA Report 832 (1945).

for which the streamline geometry appears most reasonable because this solution is most critical to later calculations. The middle streamline shown in Figure 4 was chosen, and the calculated cross velocity gradient for this stagnation point was 4.343. Since this agreed well with the estimated cross velocity gradient of 4.252, this is considered a valid or permissible solution for the upstream flow. If this cross velocity gradient were not correct, another iteration would be needed for the upstream solution with a new estimate of the cross velocity gradient.

The cross velocity gradient DVOOI determines the perturbation of the stagnation streamline due to the presence of the airfoil. A greater perturbation is realized with increasing values of DVOOI.

After decision point 1, there are two possible paths for further flow calculations. The path to the left corresponds to flow integration on the lower surface. For this path, J=2 and subroutine LWRCRIT is computed. The path to the right corresponds to flow on the upper surface. For this path, J=1 and subroutine UPRCRIT is calculated. Decision points 2 and 3 determine whether subcritical or supercritical flow options are to be taken on the upper or lower surface. A simple test was made for selecting the supercritical or subcritical options. This information is stored in ICRIT(J). Thus ICRIT(1) = 1 for supercritical flow on the upper surface and ICRIT(2) = 2 for subcritical flow on the lower surface. Once a decision on flow criticality has been made, flow integration may proceed to the flow solutions on either the upper or lower surface.

From decision point 2, the next step in flow calculation is subroutine LWRINIT, the initial solution on the lower surface. Depending on decision point 1, there are two possible paths for further flow integration. The path for ICRIT(2) = 1 is invalid since in its present form, the program is not prepared to handle supercritical flow on the lower surface. For ICRIT(2) = 2, flow integration is further computed by subroutines SUBCRT1 and SUBCRT2 which calculate subcritical flow. The output of these three subroutines is shown in Figure 5. For a permissible solution, the calculated Mach number along the airfoil surface should return to a value fairly close to the free-stream Mach number of 0.729. Decision point 4 consists of determining an appropriate value for

CYD. Inspection of Figure 5 shows that the value of CYD = 0.8194 is appropriate. Here the Mach number increased to a maximum at midchord and decreased to a value of 0.71 at the trailing edge.

An appreciation of the physical significance of the parameter CYD requires knowledge of the stagnation streamline geometry given in Figure 6. The control volume is the one outlined by points b, d, and f. Points f and d represent values which were computed in the upstream integration. The mass flow into the control volume is normal to the line d-f. Since there is no mass flow through the stagnation streamline or normal to the airfoil, the mass flow out of the control volume is normal to line b-d. Hence the mass flow out of the control volume is fixed and is equal to the area under the curve in Figure 7. The ordinate ρV is the mass flux across the line b-d, and the abscissa n is along the line b-d normal to the airfoil. Depending on the value of CYD, the product $\rho_b V_b$ can take on several values. Hence the velocity at the initial point on the airfoil V_b can be varied according to CYD. The initial velocity decreases as CYD increases.

Once an appropriate solution has been found for the lower surface, IGO(J) is set equal to 1 and control is transferred to decision point 5. Both IGO(1) and IGO(2) must equal 1 in order to proceed to DWNSTRM; otherwise control is transferred to decision point 1 and the other path is chosen for flow integration.

In the case discussed so far, the lower surface has already been computed and the upper surface flow remains to be computed. Upper surface flow has been assumed to be supercritical and control can be transferred to subroutine UPRINIT. After initial conditions in subroutine UPRINIT have been calculated, one of two paths can be chosen for upper surface flow integration, depending on the value of ICRIT(J). If ICRIT(1) = 2, the flow is assumed to be subcritical and further flow integration proceeds in the same manner as discussed previously. If ICRIT(1) = 1, the flow is assumed to be supercritical and control is transferred to SPRCRT1. Subroutines UPRINIT and SPRCRT1 compute the initial flow solution on the upper surface. The varying parameter for flow integration is CYDU. Decision point 6 is concerned with determining a value for CYDU so that

the velocity gradients are continuous through the sonic point. The graphed output of this iteration is shown in Figure 8. A value of CYDU = 1.074974 determines continuity of the velocity gradient through the sonic point and is a satisfactory solution for decision point 6.

Once the initial solution has been completed, calculation of the flow integration is undertaken for the upper surface including the effects of the shock foot. The appropriate value of CYDU has already been determined in subroutine UPRINIT and the flow should return to near free-stream values if the stagnation point and the shock location have been chosen judiciously.

In some cases it may be desirable to modify the flow solution during some intermediate step. The velocity distribution along y which is output from one step may not be appropriate, and some adjustment of the y-component velocity calculated near the airfoil surface may be made by using a Lagrangian or a parabolic curve fit along the y coordinates.

Once again, decision point 5 is encountered and since both upper and lower surfaces have been computed, control can be transferred to subroutine DWNSTRM. Subroutine DWNSTRM is concerned with the calculation of downstream flow conditions. If the value of SL (the shock location on the upper surface of the airfoil) is correct, flow will return to near free stream values. If this value is incorrect, subroutines SPRCRT2 and DWNSTRM must be reiterated with varying values of SL. The results of such an iteration process are shown in Figure 9. The downstream flows based on two shock locations should bracket the free-stream value ten chord lengths downstream from the body ($P/P_{\infty} = 1$ at $x/c = 10$). As shown in Figure 9, the exact shock location lies between $x/c = 0.57$ and 0.58.

When the downstream flow conditions most nearly approximate free-stream values for the upper surface, parameter CYDL can be varied for the lower surface to find the value which most nearly approximates free-stream flow conditions downstream of the airfoil. In this case, subroutines LWRINIT, SUBCRT1, SUBCRT2, and DWNSTRM are iterated to find a value for

CYDL which most nearly approximates free-stream conditions downstream of the airfoil. The results of this iteration process are shown in Figure 10. The downstream flow conditions most nearly approximate free-stream values at CYDL = 0.8131, and this value of CYDL is chosen to compute the lower surface flow conditions.

There is one remaining step in the solution process, namely, to check the calculated pressure distributions and determine whether the Kutta condition is met at the trailing edge. The upper and lower surface pressure distributions are shown in Figure 11. Since the pressures calculated at the trailing edge for upper and lower surfaces have less than 3-percent error, the assumed stagnation point is correct. If these pressures had not matched at decision point 9, a change would have been required for the stagnation point and the solution process would proceed again from decision point 2.

The upper surface pressure distribution depends greatly on the value of β, the oblique shock angle of the shock foot. The shock location moves forward with decreasing values for β. In this particular example, a change of entropy was allowed through the shock wave and the angle of β was assumed to be 70 deg.

Other Airfoils

The procedure for calculating the transonic flows over other airfoils is basically the same as above except that a change has to be made in subroutine ARFL. An analytic function does not exist for airfoils other than NACA 4-digit series, and some method of airfoil representation must be used. The method used for this program is the spline fit (see Appendix A). The method requires a given set of data points and the first derivatives at the beginning and end points of that set. The coordinates of the airfoil should be very accurate for a smooth curve fit. It is possible to find an airfoil shape with a smooth second derivative fit by varying the beginning and end slopes of the airfoil.

Figure 12 shows the fitted curve for a particular airfoil, and a plot of the second derivatives for this curve. The smooth fit for the second derivatives ensures that the airfoil curvature is pretty well represented.

APPLICATION OF INTERACTIVE GRAPHICS

The subroutines previously described have been incorporated into an interactive graphics program so that program execution can be accomplished most efficiently. The interactive graphics program has been written with the help of Graphic Pac, an NSRDC-developed software package for use with graphics facilities.* The Graphic Pac features include virtual memory data management for both graphic and nongraphic data and a comprehensive collection of interactive facilities; program control is modified during execution by the use of subroutine WAITE. When a call is made to this subroutine, execution stops and the program awaits input from an attention source. Attention sources are the light buttons and text entities which appear on the screen, and these may be signalled by the light pen.

When Graphic Pac is used, all subroutines have to be compiled into a relocatable binary format by PRELOAD, an NSRDC-developed utility.** Once the graphics program and the subroutines have been compiled by PRELOAD, they are loaded into a new task format by TSKLOAD, another NSRDC-developed utility program. It is the TSKLOAD format which is executed. When this program is loaded by using IGSGO, it makes nominal demands on the CDC 6700 computer. The control cards needed to create the taskload file are shown in Figure 13, and those required to make a graphics run are shown in Figure 14.

---

*Reported informally in NSRDC Technical Note CMD 42-28 (Graphic Pac — A Subroutine Package for Interactive Graphic Application Programming), August 1973.

**Reported informally in NSRDC Technical Note CMD 51-72 (PRELOAD — A Binary Deck Library Loader for the CDC 6700 Computer), October 1972.

The CDC 6700 central processor should be specified to compile and load the program most efficiently. During loading, the program uses approximately 400 CPU sec, has a field length of 110000 Octals, and resides in central memory for about 1 hr. During execution, the program has a field length of 20000 Octals.

The structure of an interactive graphics program is somewhat different from a program used in batch processing. In order to have maximum control over the program and to allow input changes when necessary, there are many points in the program where program execution pauses and waits for a signal from one of the attention sources. An attention source can be a light register used to type in new input information or an asterisk used to signal execution of a new batch of coding. The flow chart in Figure 15 indicates the possible paths for program execution. The nodes indicate possible input changes.

Each of the tasks in the program perform a well-defined function. Half of them display information calculated by a MIR subroutine and the other half maintain the screen displays. A brief description of each task is given in Appendix D. The subroutines used by these tasks and their functions are given in Appendix B.

According to Figure 15, there are many possible paths for the program to follow. However, it is not necessary to use all these paths in the solution process. In some cases, a decision box could have been used instead of a node. In order to avoid a complex logic diagram, however, the format of Figure 15 was chosen. This figure at least gives an indication of the versatility of the interactive graphics program which allows many possible paths instead of two or three from a particular program control point.

BASIC FORMAT

Figure 16 gives the basic format for the graphic output of a step. Most of the screen display is given to the plot of currently computed output. Sometimes two plots may appear in this area of the screen. If for any reason at all, it is impossible to perform the integration

at this step, a large X will cover the graph display; if only a partial integration is possible, the message INTEGRATION INCOMPLETE will flash on the screen. There are two columns of light registers in the lower right-hand corner of the screen; the first gives information on flow conditions at the currently computed step and the second contains the input variables. The variables are light pen detectable, and the values in them can be changed. A current value can be erased and replaced with a blank by touching a light register with a light pen and depressing the handle of the pen. A new value can then be inserted by typing it in on the keyboard and pressing the keyboard release button. When the COMPUTE button at the bottom of the column is touched, the program will attempt to execute the step with the current input. The asterisks surrounding the airfoil in the lower left-hand corner signify the steps of the flow solution; they are coded in Figure 16. The currently computed step is identified by a flashing asterisk. The asterisks will appear only when the program is ready to execute the program step which they represent. Program control can be transferred to any other step by signalling the appropriate asterisk with the light pen. The program can be terminated at any time by using the light pen to signal the STOP button in the far left-hand corner.

The input to the graphics program consists of 32 flow solution parameters and a maximum of 40 airfoil data points and the first derivatives at these data points. These airfoil data points and their first derivatives have been chosen to ensure a smooth second derivative curve fit in accordance with Appendix A. A description of the input data is given in Appendix E. Many of the flow solution parameters assume the values suggested in the appendix.

ILLUSTRATIVE EXAMPLE

Just as an illustrative example of flow past an NACA 0015 airfoil was used to describe the MIR program subroutines, an illustrative example of flow past an advanced transonic airfoil will describe the use of IGS. The first display to appear on the screen is that shown in Figure 17.

The four light registers contain the free-stream flow conditions and certain initial conditions:

ALPHA	= Angle of attack
MACH NO.	= Mach number
YI(UPR)	= Location of outermost strip in upper surface (given in chord length)
YI(LWR)	= Location of outermost strip in lower surface (given in chord length)

If these flow conditions are satisfactory, control may be transferred to the first step in the flow solution by signalling the light button PROCEED.

The first step in the solution is the calculation of the upstream flow conditions. The necessary parameters for the upstream solution are the number of strips used in integration and XOO, the distance from free-stream flow conditions to the stagnation point on the airfoil. The parameter NN indicates the number of strips used for the bulk of integration, and NA indicates the number of additional strips used in the vicinity of the airfoil. For greater accuracy, it is recommended that eight strips be used in the vicinity of the airfoil. Figure 18 indicates the screen display corresponding to this solution.

The flashing light ahead of the airfoil in the lower left-hand corner of the screen display indicates that the upstream solution is ready for execution. When the COMPUTE light button at the bottom of the second column of light registers is signalled, this step will be executed by using the input values currently in the light registers. The computed values of YSO and DE will be displayed in the first column. These values should be less than 0.1. The graphic output shows Y versus M, the velocity profile at the final station of upstream integration, and M_o versus X, the variation of Mach number along the stagnation streamline. These two graphs are characteristic of an appropriate solution.

If the computation of the upstream solution is complete, the program may proceed to the stagnation solution. The necessary parameters used to

iterate on this stagnation solution are XS, the X-coordinate of the stagnation point, YSO, the distance that the stagnation streamline is perturbed by the airfoil, and DVOO(I), the cross velocity gradient used in the upstream solution. The screen display is shown in Figure 19.

If DVOO(F), the cross velocity gradient calculated at the particular stagnation point, does not agree with DVOO(I), then DVOO(I) must be changed to the newly calculated value, and the upstream solution must be recalculated. By signalling the light far to the left of the airfoil, control is transferred back to the upstream solution which is computed by using the new DVOO(I). When the streamline geometry seems reasonable and the cross velocity gradients agree at the stagnation point, the program may proceed to the step which determines flow criticality on either the upper or lower surface. For example, consider the flow on the upper surface. Control is transferred to this step by signalling the light just above the leading edge of the airfoil.

The screen display shown in Figure 20 determines the type of flow present on the upper surface. In this case the Mach number reaches a value of 0.96 in a relatively short distance, and so it is safe to assume that supercritical flow is present on the upper surface. By signalling the SUPERSONIC light button, program control is transferred to the next step which computes the initial conditions on the upper surface.

The screen display of Figure 20 also indicates the light button LAGRANGIAN. When it is signalled, the normal velocity component at the innermost strip at the initial step will be corrected using a Lagrangian curve fit. The light button LAGRANGIAN will disappear and the light button PARABOLIC will appear in the same area on the screen. Similarly, when the latter is signalled, the normal velocity component at the innermost strip at the initial step will be corrected using a parabolic curve fit. If neither light button is signalled, the normal velocity component at the innermost strip will not be modified during the flow integration.

The necessary parameters for calculation of the initial solution are XA, the initial point of flow integration, and CYD, which determines the initial velocity profile shape. The screen display of Figure 21

illustrates the iterative process used to satisfy the flow conditions in this step. After an initial point has been chosen, the parameter CYDU is varied until the velocity gradient DUDX is continuous through the sonic point.

The following additional information is included to help proceed to a converged solution. The value of RBUB ($\rho_b V_b$ in Figure 7) should be less than 1.1 and CYDU should be increased until this requirement is met. If CYDU is too large, the velocity gradients will become negative and the flow will become subsonic, prohibiting further integration. If the solution still does not converge, the number of strips NN should be decreased by one. When an appropriate CYD value is chosen and integration is completed, a light above the airfoil will signal that the upper surface airfoil solution is now ready to be computed. Further refinements to the initial solution can now be made or control can be transferred to the next step by signalling the light above the airfoil.

The screen display of Figure 22 indicates the airfoil solution on the upper surface. The location of the shock foot should be chosen so that flow integration may proceed from the initial solution to the trailing edge of the airfoil. If the shock location is chosen too close to the nose of the airfoil, the flow will accelerate to supersonic again after the shock wave, prohibiting further integration; if the shock location is chosen too close to the trailing edge of the airfoil, the flow becomes over expanded before the shock foot, prohibiting further integration. A careful choice of shock foot will allow integration to proceed to the trailing edge.

If the solution on the upper surface is completed, control may be transferred to the step which determines flow criticality on the lower surface by signalling the light under the leading edge of the airfoil. The screen display for this step is shown in Figure 23. Since the local Mach number is below 0.6 for at least 5 percent of the airfoil surface, it is safe to assume that subcritical flow exists on the lower surface of the airfoil. The program now proceeds to the step which computes the airfoil solution on the lower surface.

The screen display of Figure 23 also shows the light button LAGRANGIAN. Correction to the innermost strip y-component velocity can be made by signalling this light button in the same manner as indicated for Figure 20.

The screen display for the airfoil solution is shown in Figure 24. The parameters for this solution are the same as for the initial solution on the upper surface. If the chosen value of CYDL is too small, the message FLOWS NOT MATCHED will appear where UB = 0.699485 now appears on the screen. When the value of CYDL is increased, the value of RBUB will decrease and flow integration may proceed. A particular choice for CYDL will allow integration to proceed to the trailing edge. Further improvements can be made to the airfoil solution or control may be transferred to the downstream solution. The upper surface and lower surface can be computed in any order, but the downstream solution cannot be computed until both upper and lower surfaces are computed.

The screen display of Figure 25 will appear when the light to the right of the airfoil is signalled. A satisfactory solution for this step would be one in which the graph of PO versus X has values fairly close to one, meaning that the computed pressures are fairly close to free-stream pressures downstream. Since control was transferred to this step from the lower surface, the downstream solution considers the flow regime from the slip streamline to the outermost strip on the lower surface. In order to find a solution which will yield free-stream flow conditions in this regime, an iteration must be made on the lower surface airfoil solution and the downstream solution by varying the value of CYDL. Once a satisfactory solution has been found, control may be transferred to the airfoil solution on the upper surface by signalling the light just above the airfoil.

The screen display for the airfoil solution on the upper surface is the same as previously shown in Figure 22. Since both upper and lower surfaces have been computed, control may be transferred to the downstream solution by signalling the light to the right of the airfoil.

Since control was transferred to this step from the upper surface, the downstream solution considers the flow regime from the slip streamline

to the outermost strip on the upper surface. In order to find a solution which yields free-stream flow conditions downstream, an iteration must be made on the upper surface airfoil solution and the downstream solution by varying the value of SL. When an appropriate solution has been found, control may be transferred to the final program by signalling the light which appears on the airfoil.

Figure 26 illustrates the screen display of the final program step. The validity of the solution can be checked by inspecting the pressure distributions computed in the airfoil solution for both upper and lower surfaces. (Pressures on the upper and lower surfaces should match at the trailing edge in order to satisfy the Kutta condition.) Inspection of Figure 26 shows that this condition has not been met (there is a 10-percent discrepancy between trailing edge pressures) and that further action should be taken to correct this situation. Program control can be transferred to the stagnation solution for the selection of a new stagnation point. Then the whole solution procedures described above should be repeated.

Solutions exist for each of the four major iteration processes which have been presented. Failure to find a bracketed solution for a particular iteration process indicates the need for further refinement in the strip arrangement of the flow field. Computational experience further indicates that special attention should be given to spline fitting the leading edge of the airfoil to ensure that the curvature of the airfoil is smoothly continuous in the region of the sonic point to avoid difficulties in attaining converged solution.

CONCLUDING REMARKS

The use of interactive graphics enables a practical application of the method of integral relations to solve transonic flow problems past lifting airfoils. For instance, 5 to 10 min of actual computer time and about 1 hr of interactive graphics time are required to determine the converged solution, i.e., pressure distribution about the given airfoil for a given flow condition. With experience, these times could be reduced still more.

Experience at NSRDC during the development of the application indicates that care must be exercised in the strip arrangement of the flow field and in the spline fitting of the airfoil coordinates, particularly near the leading edge or sonic point area in order to ensure numerical stability and accuracy.

The use of interactive graphics for this program is minimal. Further refinements might include hard copies of the output from the Cal Comp plotter. The only output now saved is that on the line printer. Use of light registers might also make it possible to keep track of previous guesses on a particular iteration.

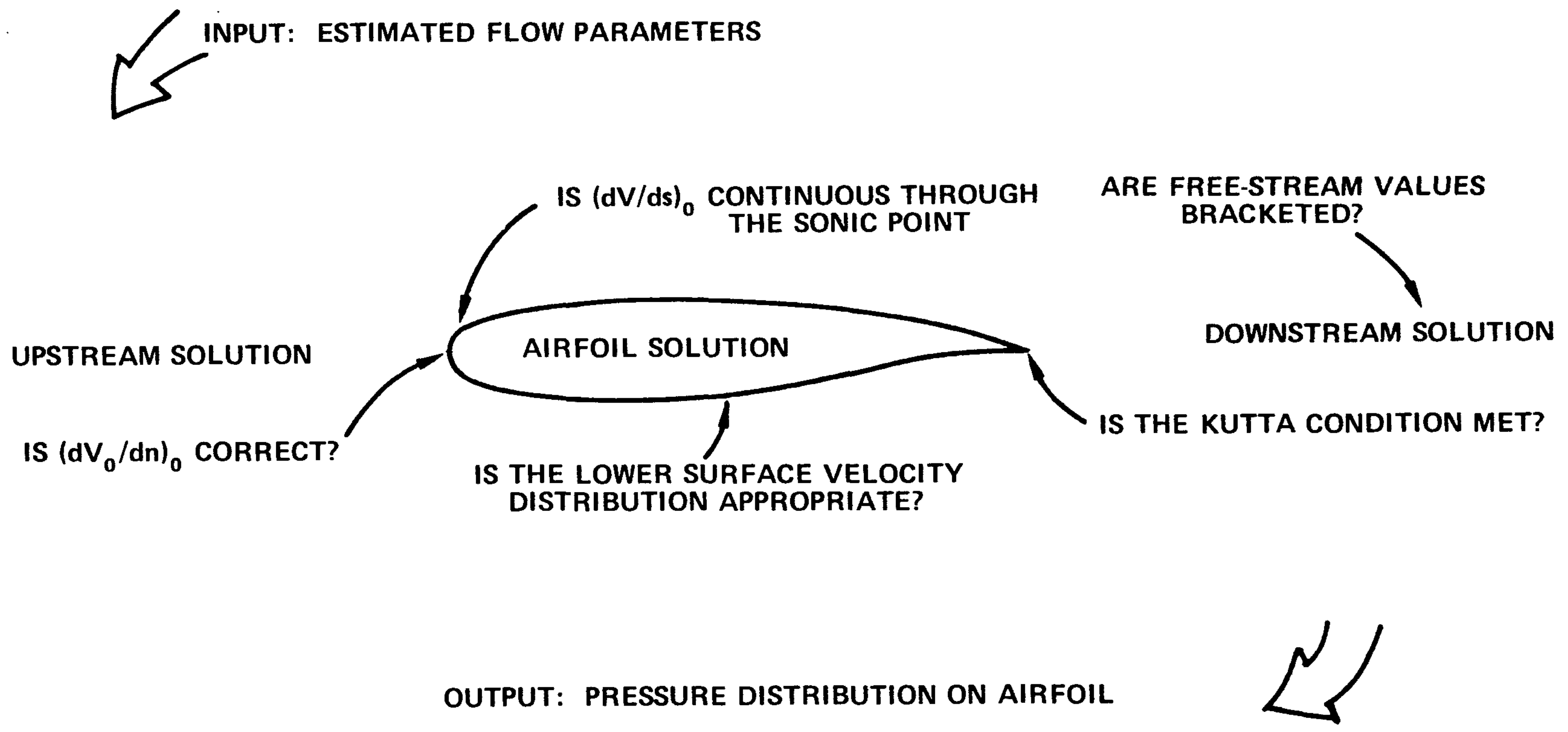


Figure 1 - The Solution Procedures for Transonic Flow Problems

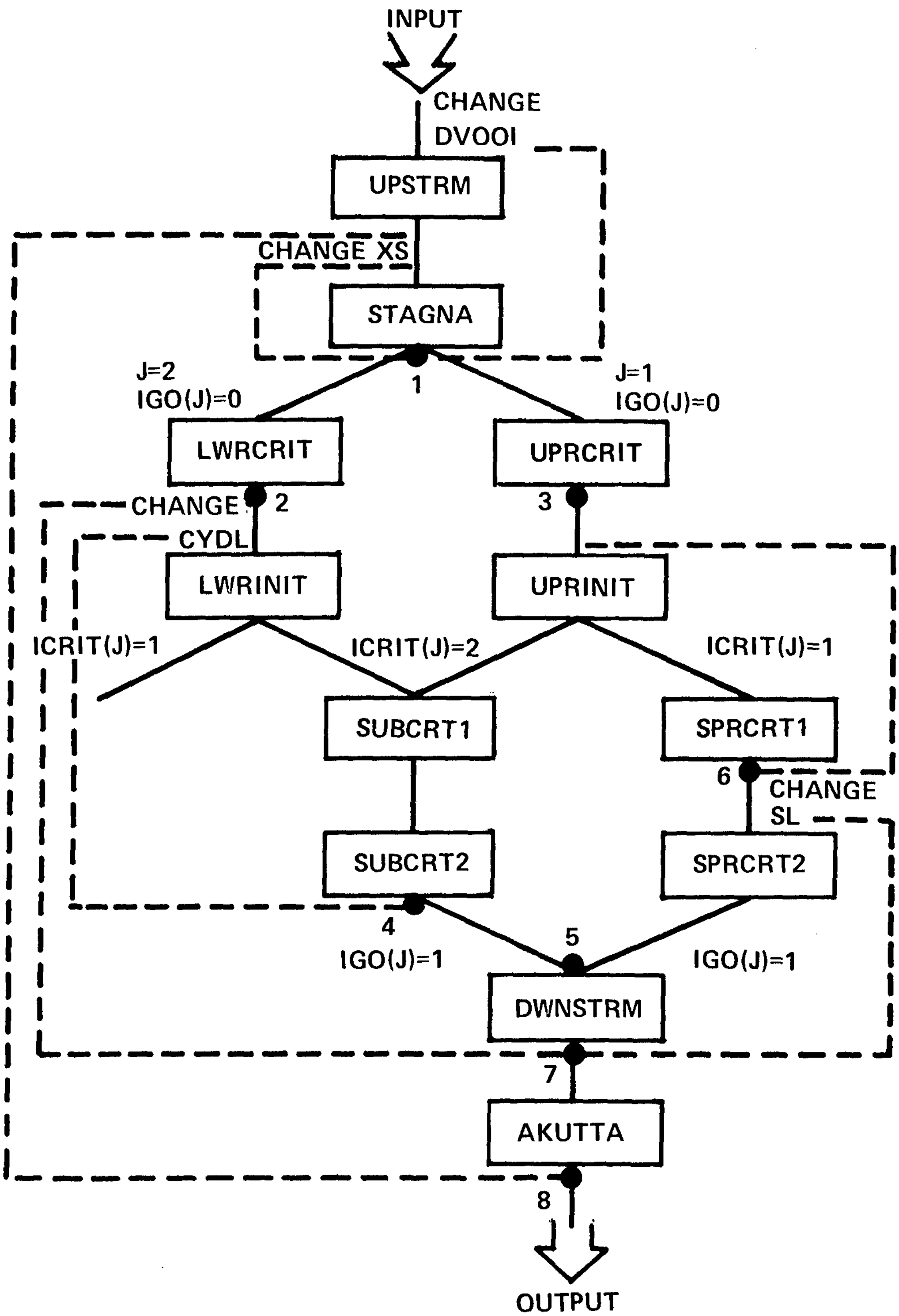


Figure 2 - Flow Chart for the More Important Subroutines

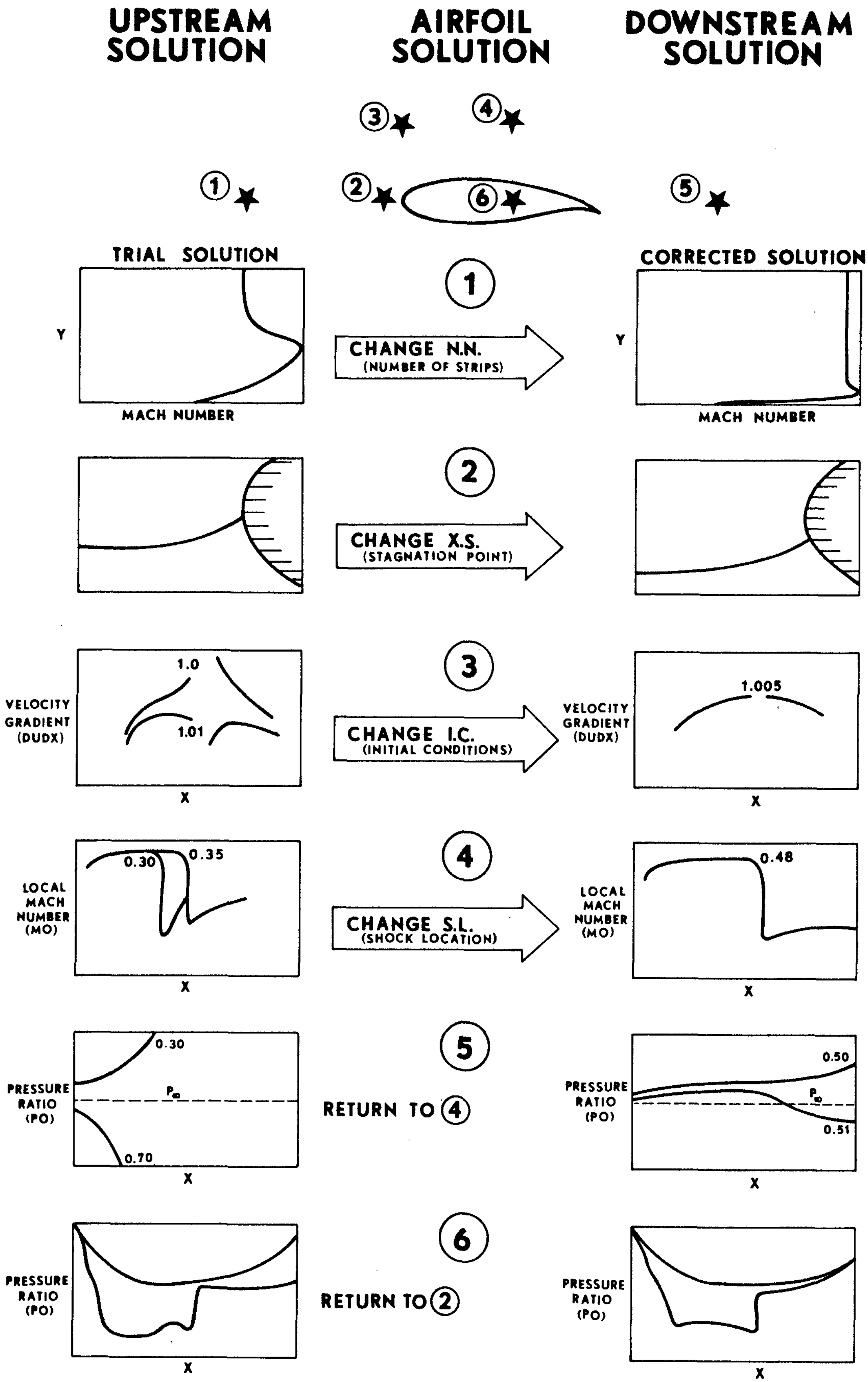


Figure 3 - Overview of the Solution Process

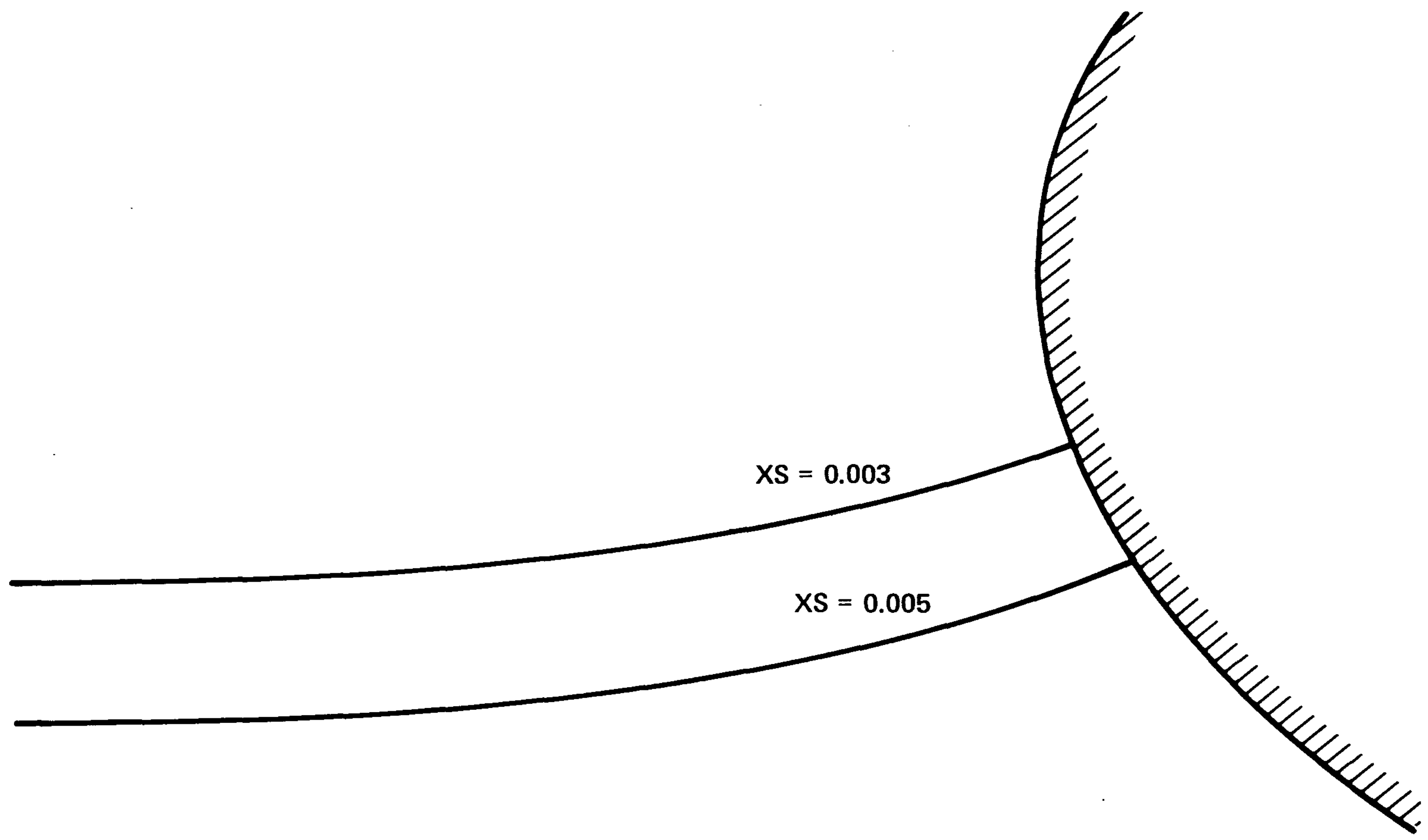


Figure 4 - Choice of Stagnation Point

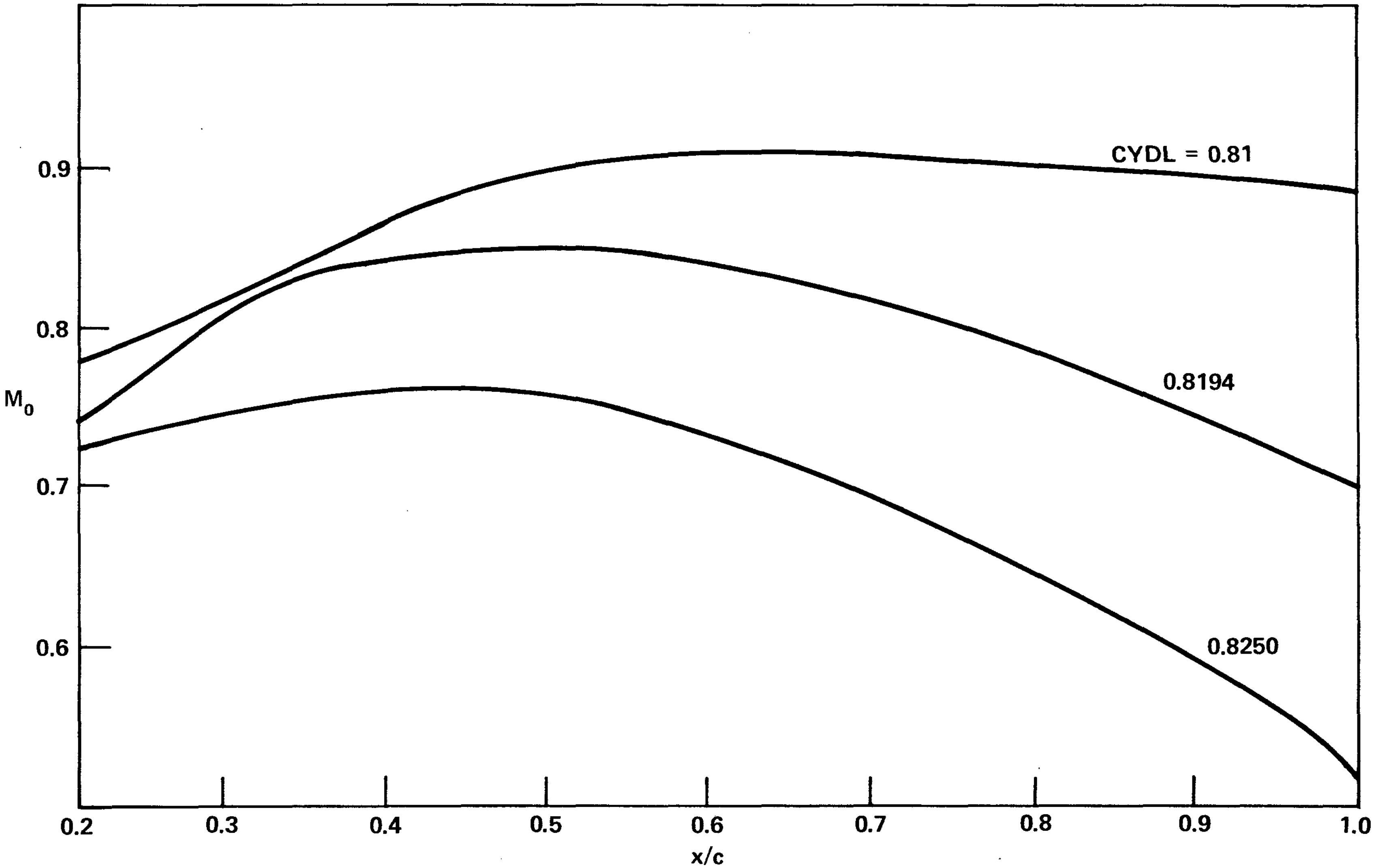


Figure 5 - Choice of CYDL for Lower Surface Flow

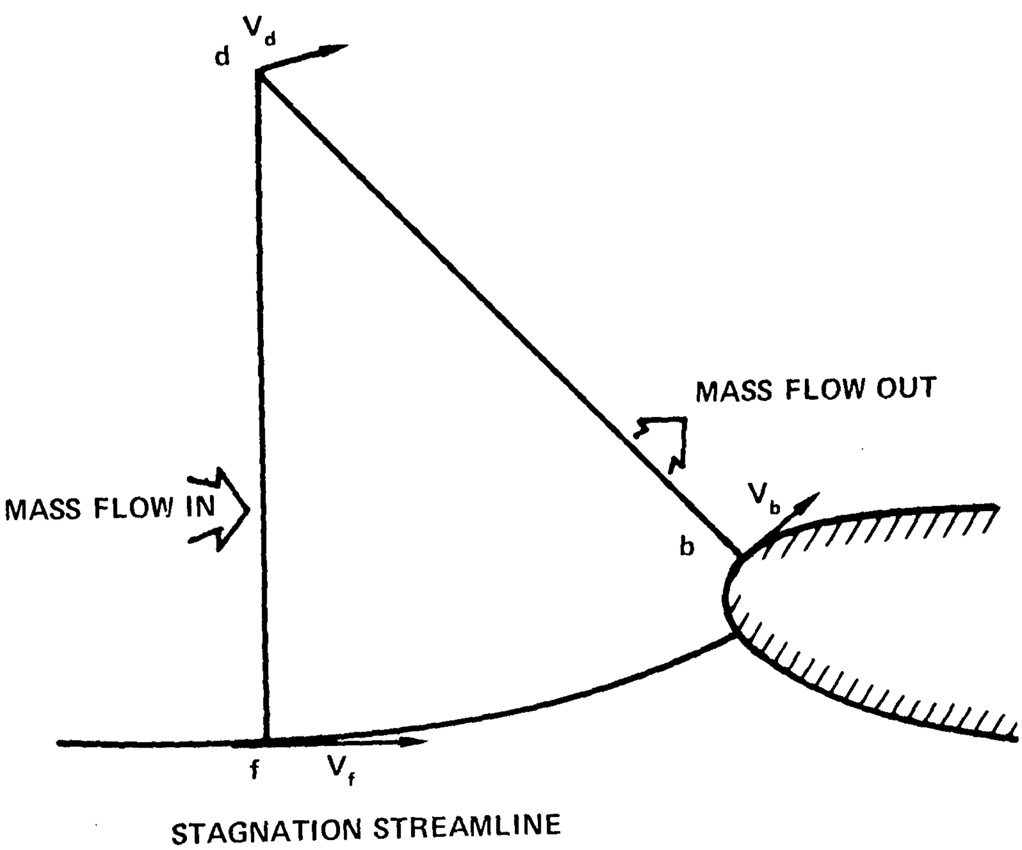


Figure 6 - Control Volume Used to Determine Initial Velocity on Airfoil Surface

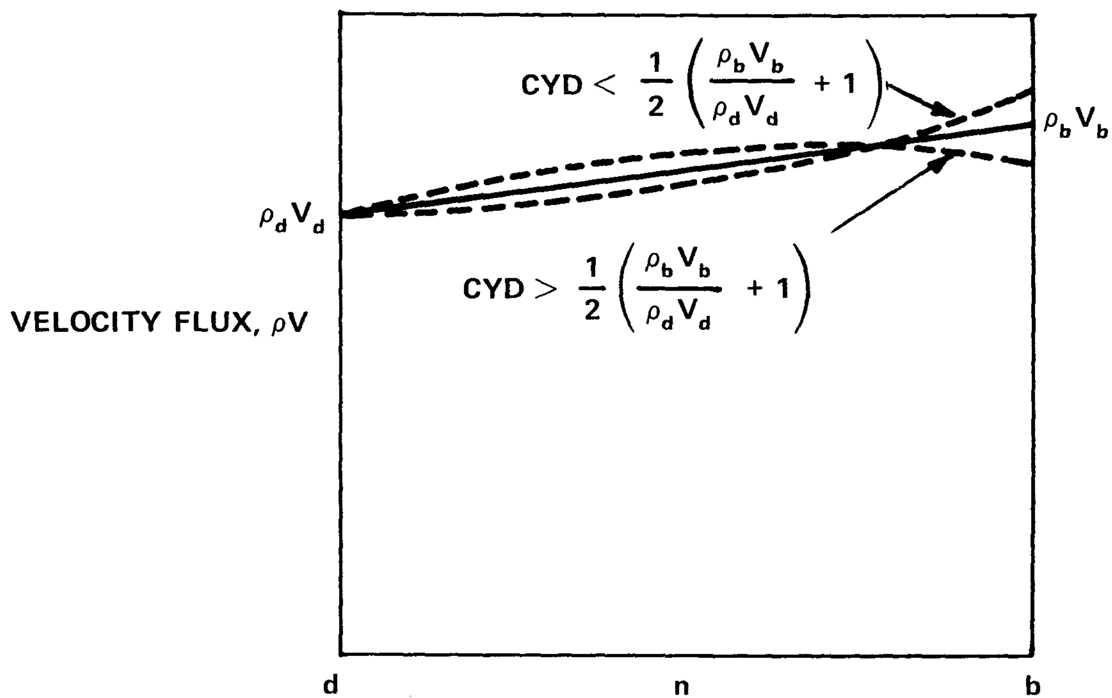


Figure 7 - Effect of CYD on $\rho_b V_b$

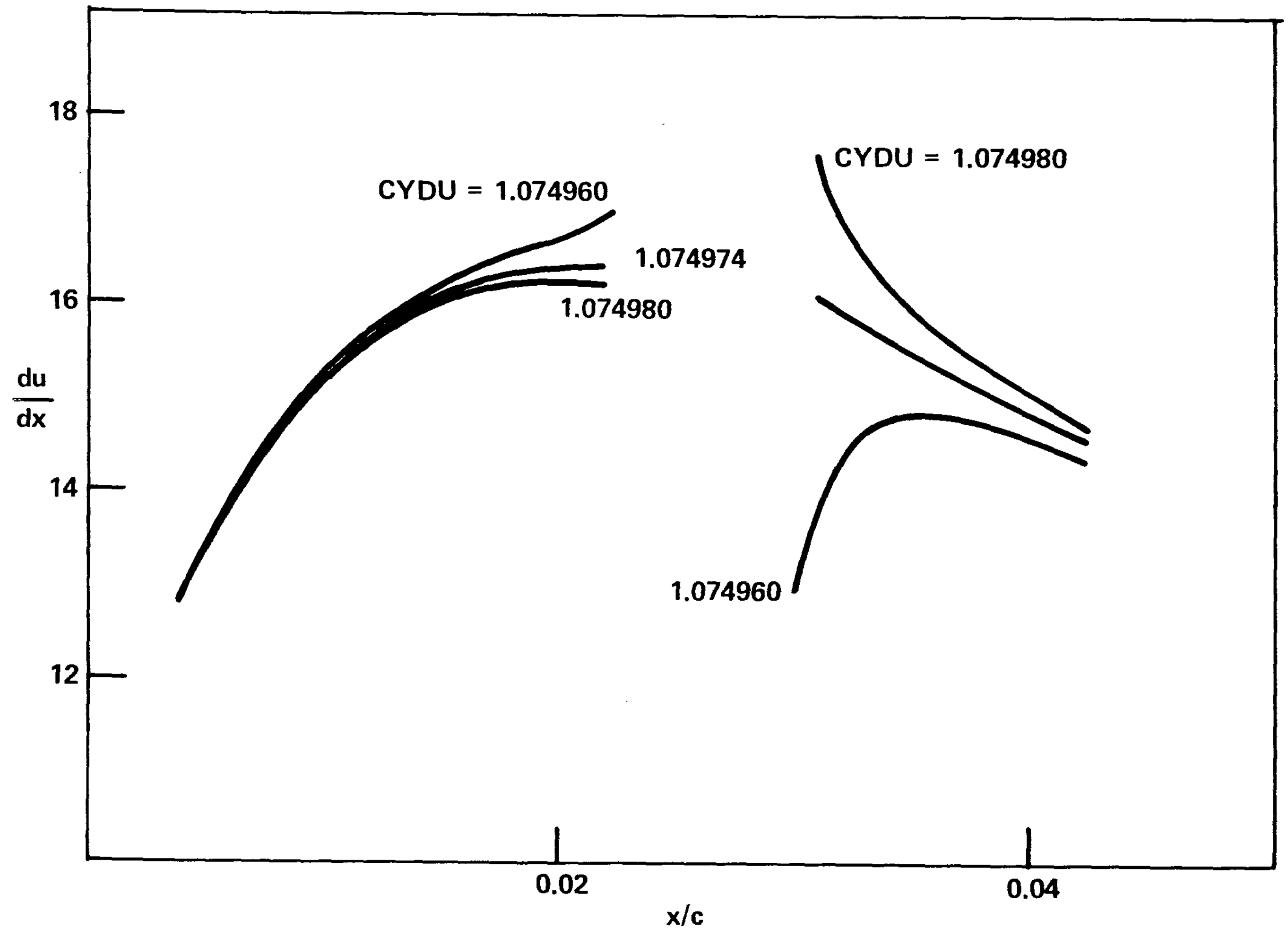


Figure 8 - Choice of CYDU for Upper Surface Flow

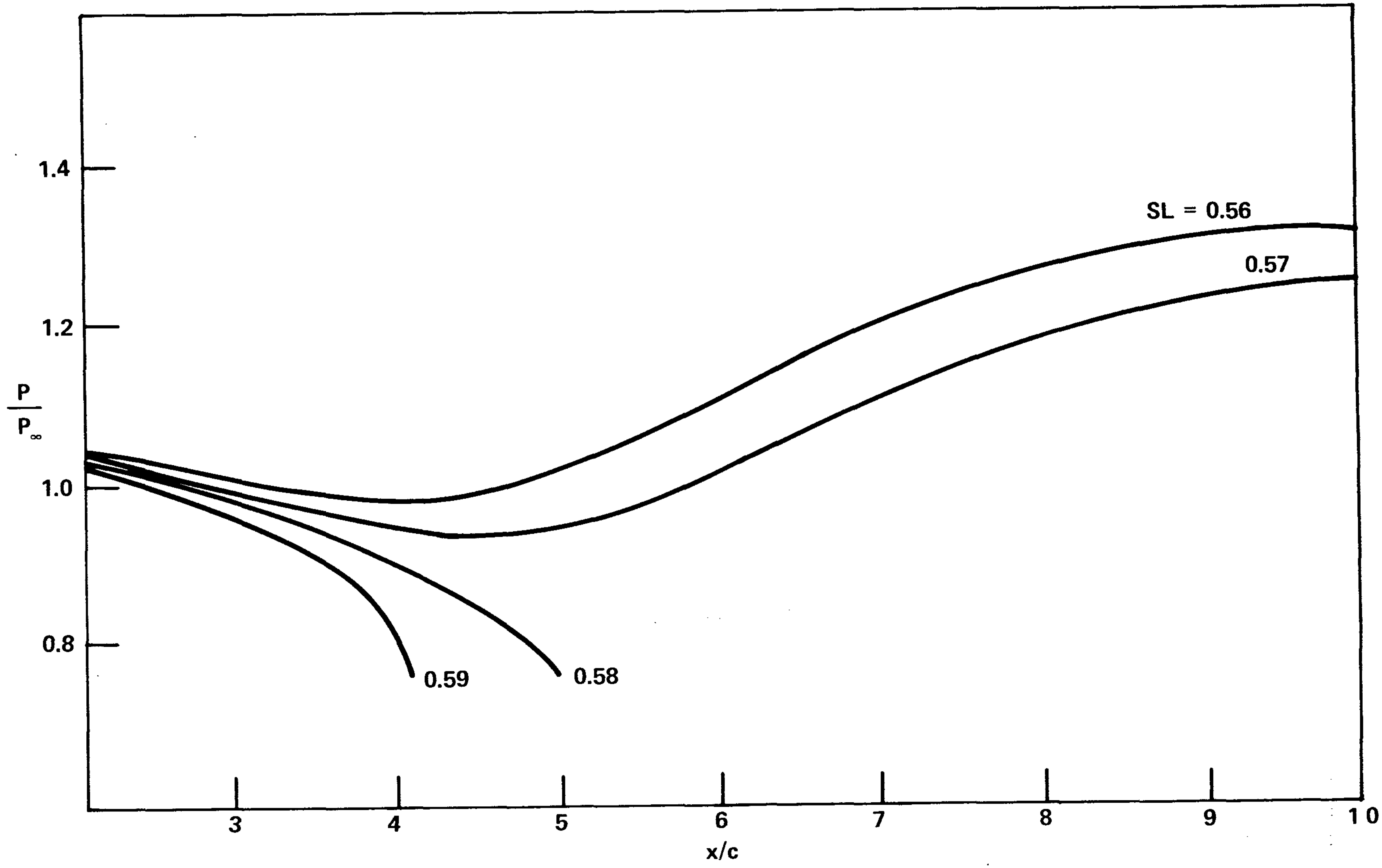


Figure 9 - Choice of Shock Location for Downstream Flow

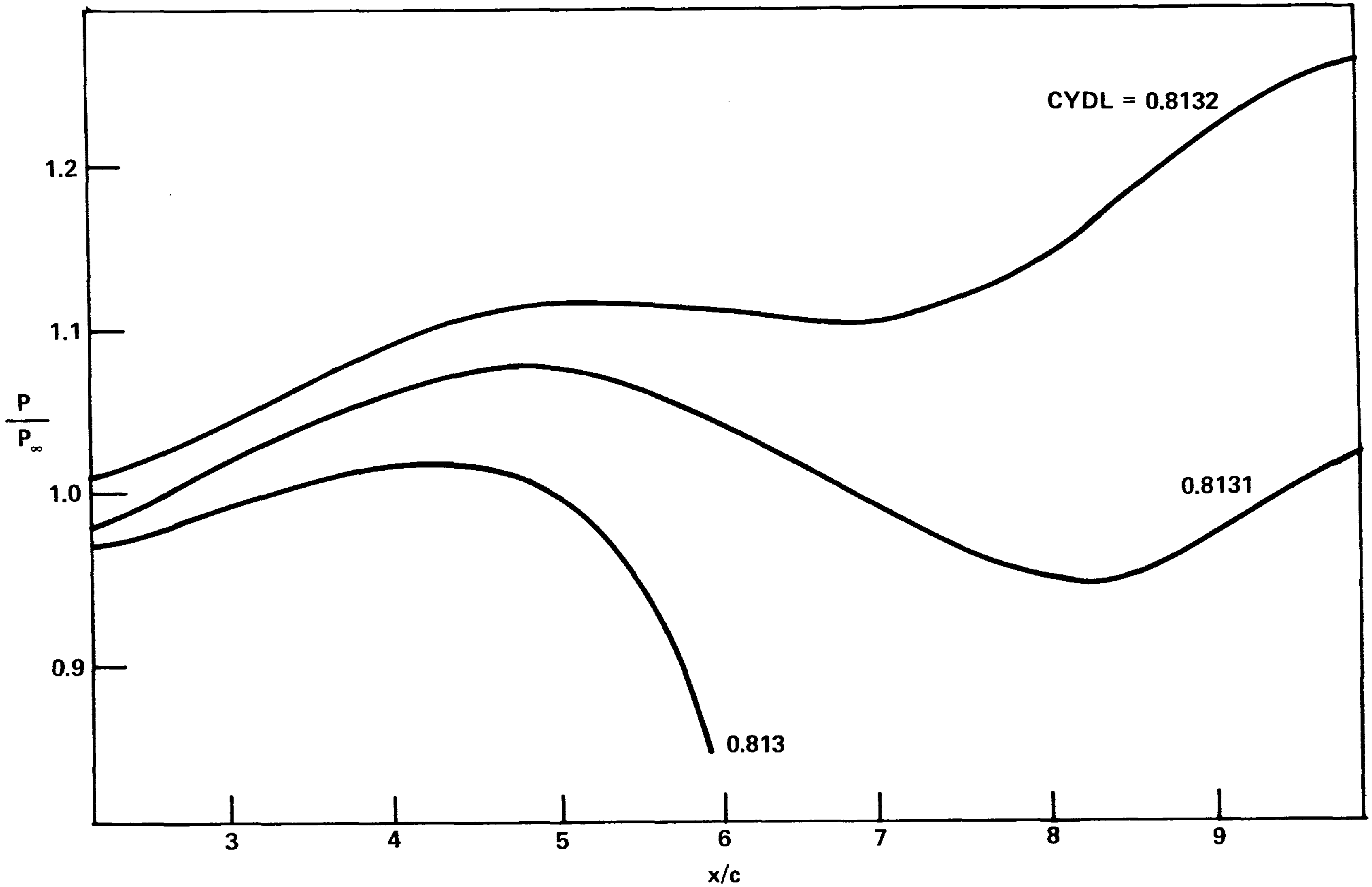


Figure 10 - Choice of CYDL for Downstream Flow

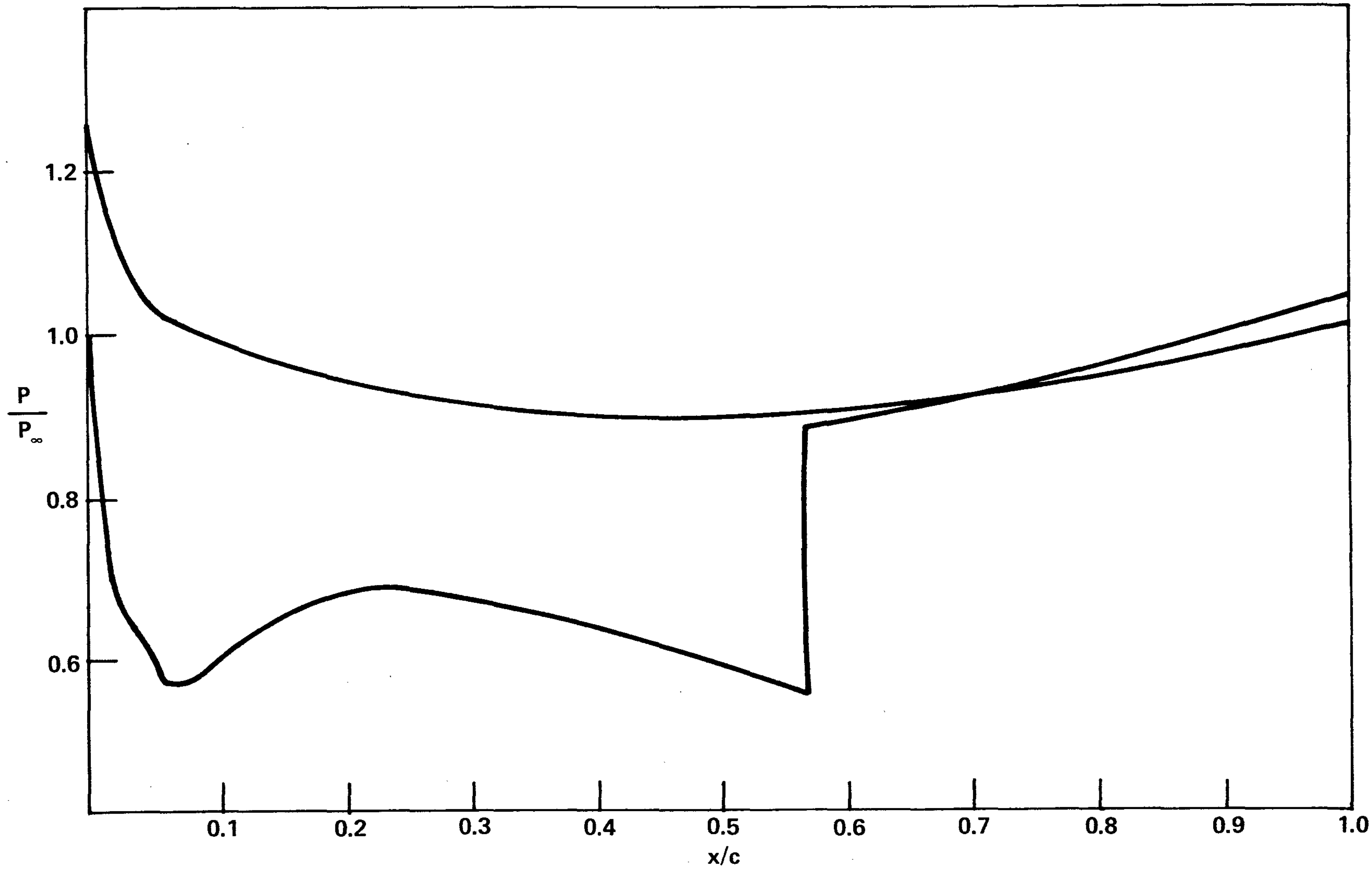


Figure 11 - Check of Kutta Condition at the Trailing Edge

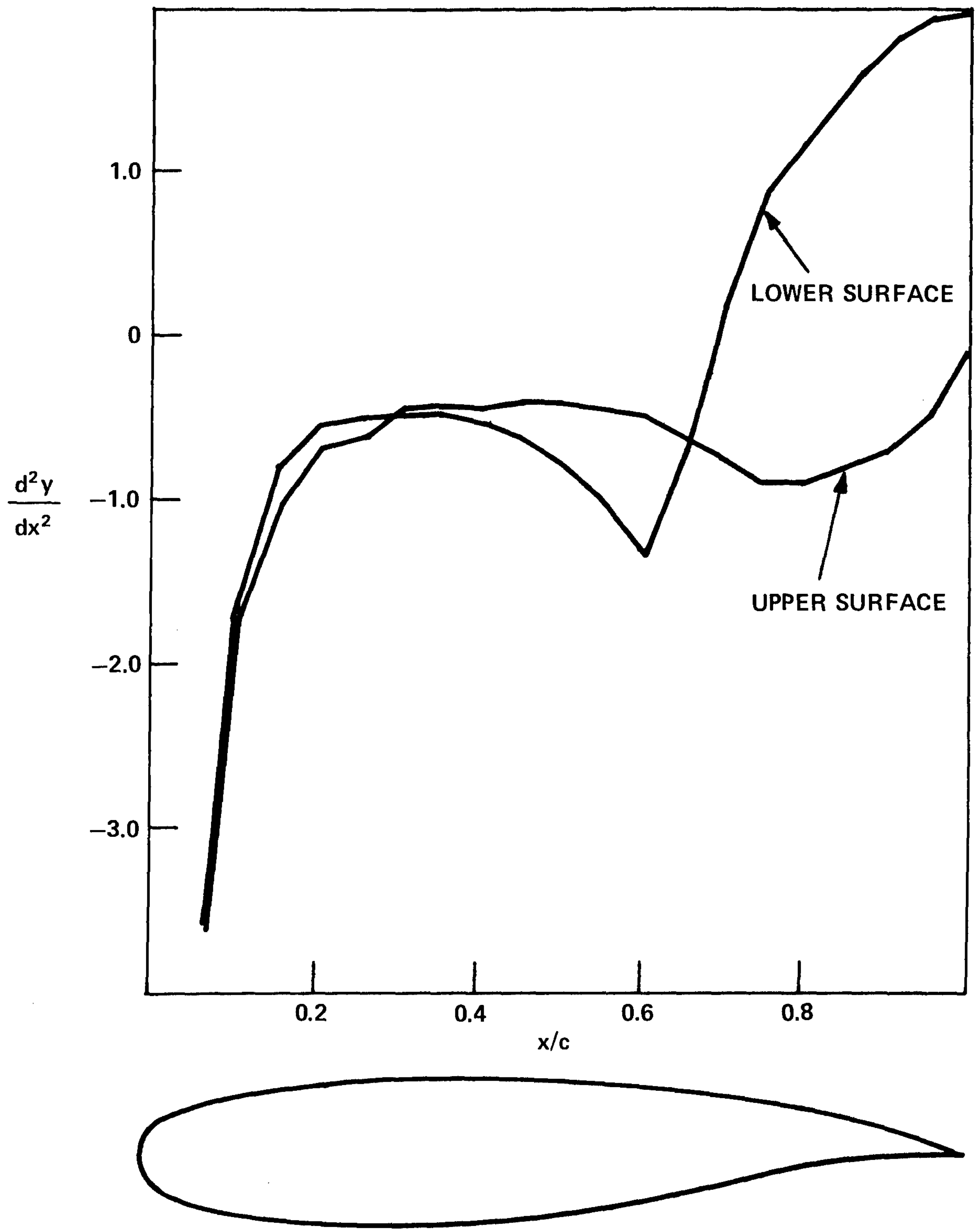


Figure 12 - Second Derivative Spline Fit for an Advanced Airfoil

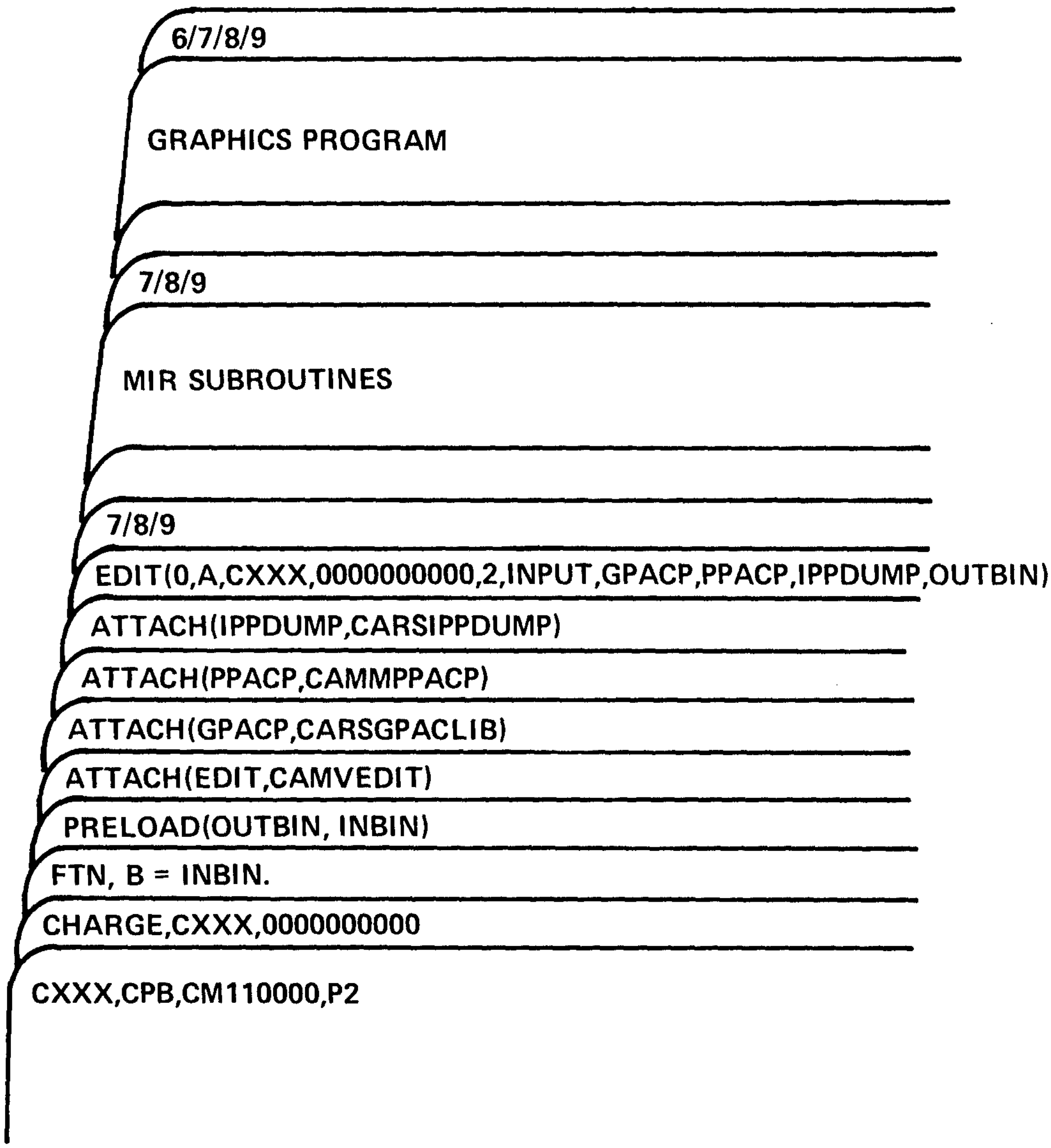


Figure 13 - Deck Setup for Creation of TASKLOAD File

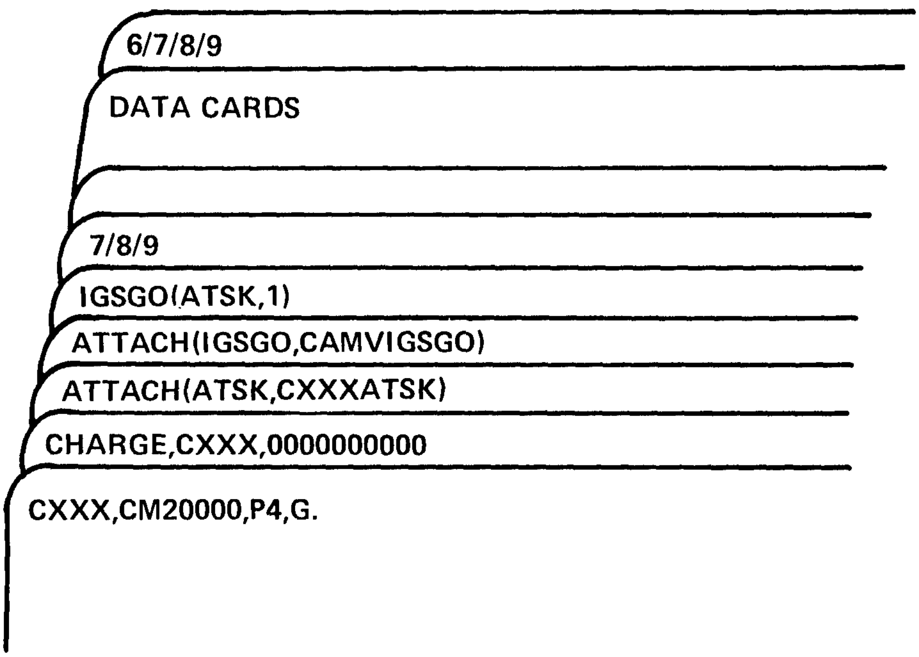


Figure 14 - Deck Setup for Graphics Run

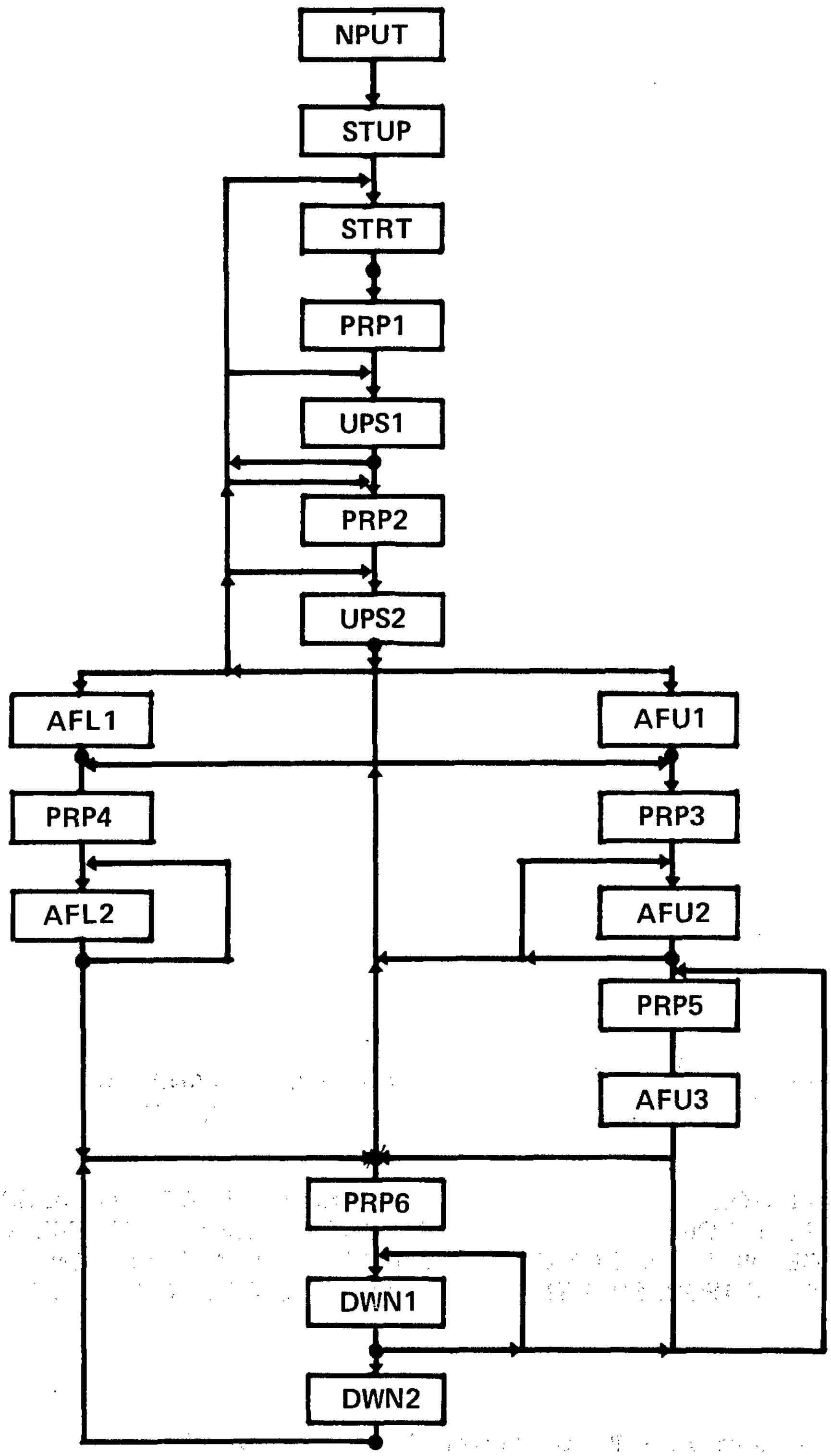
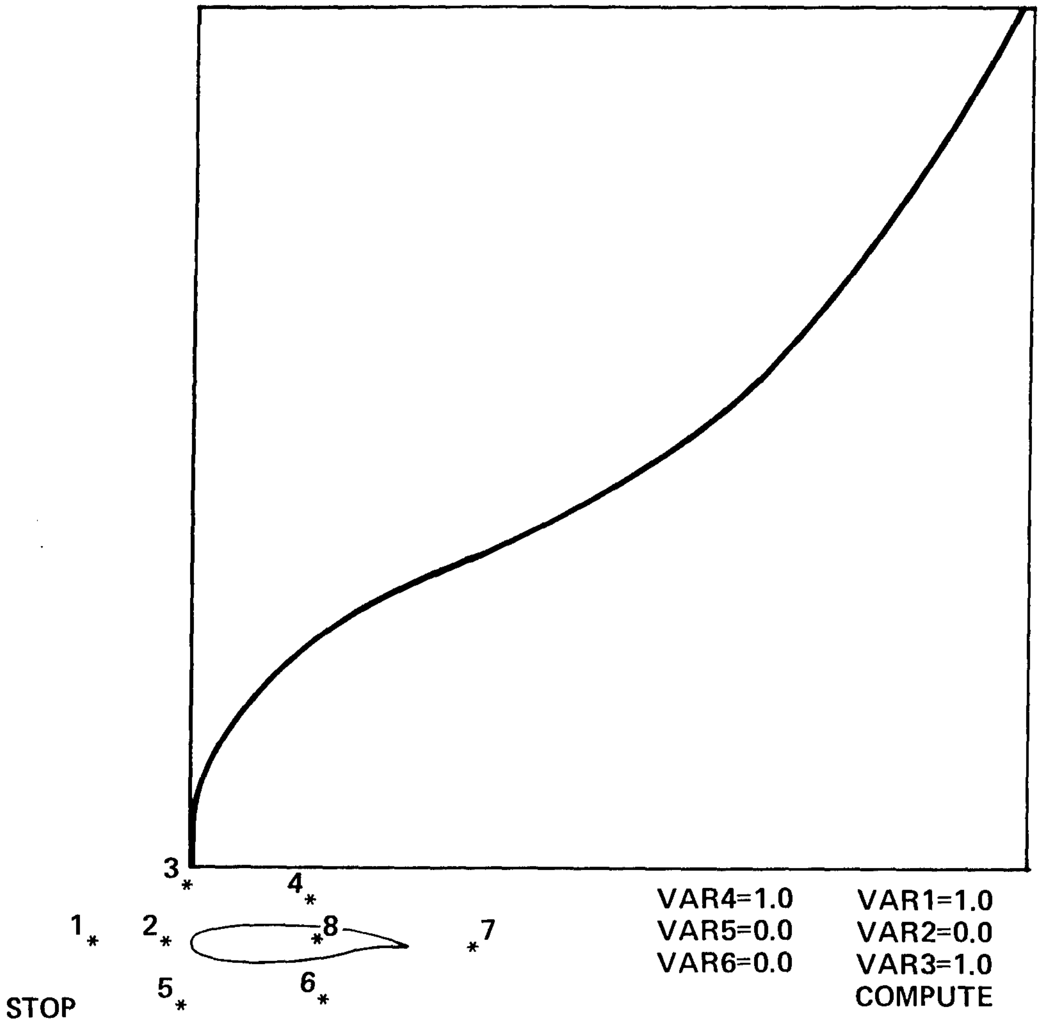


Figure 15 - Flow Chart for the More Important Tasks of the Interactive Graphics Program



1. UPSTREAM SOLUTION
2. STAGNATION SOLUTION
3. UPPER SURFACE INITIAL SOLUTION
4. UPPER SURFACE AIRFOIL SOLUTION
5. LOWER SURFACE INITIAL SOLUTION
6. LOWER SURFACE AIRFOIL SOLUTION
7. DOWNSTREAM SOLUTION
8. KUTTA CONDITION CHECK

Figure 16 - Basic Format for Screen Displays

Figure 17 - First Screen Display

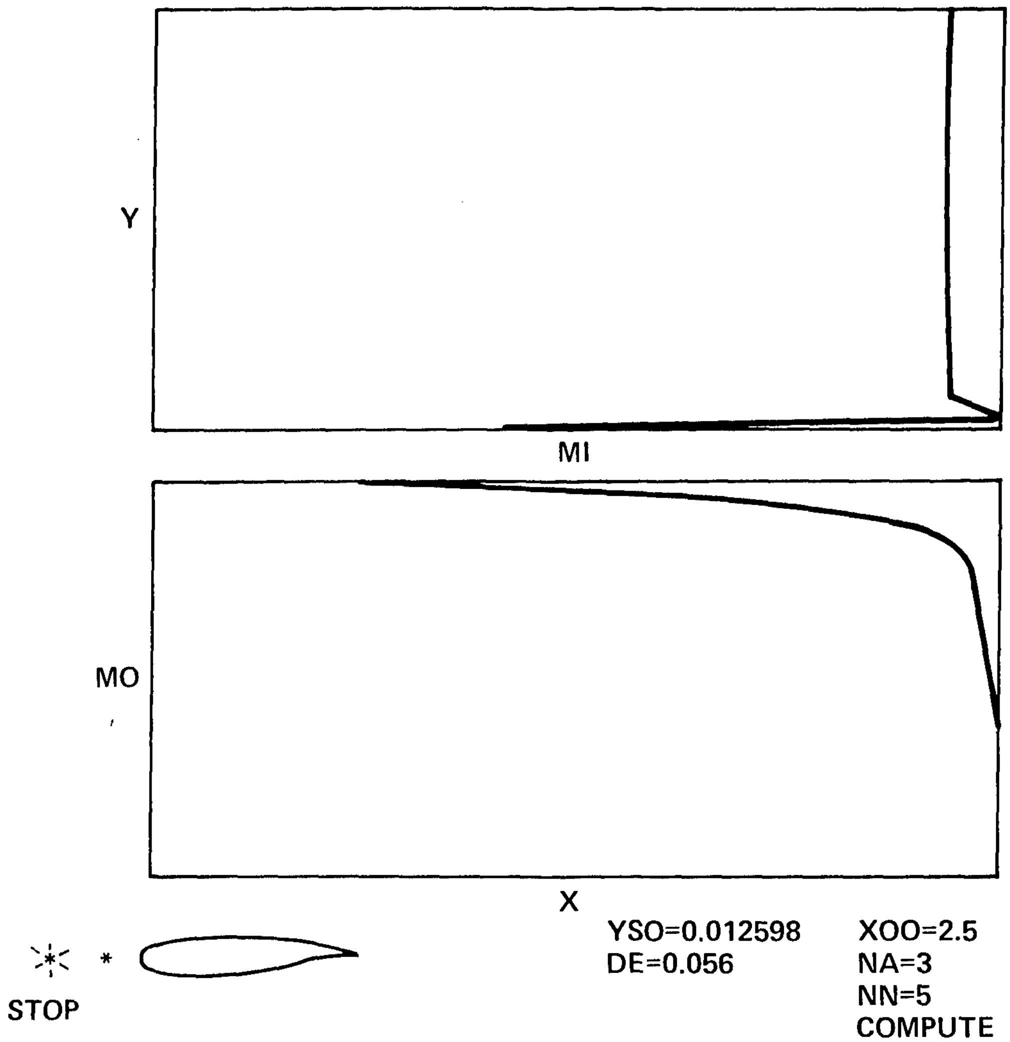


Figure 18 - Screen Display for Upstream Solution

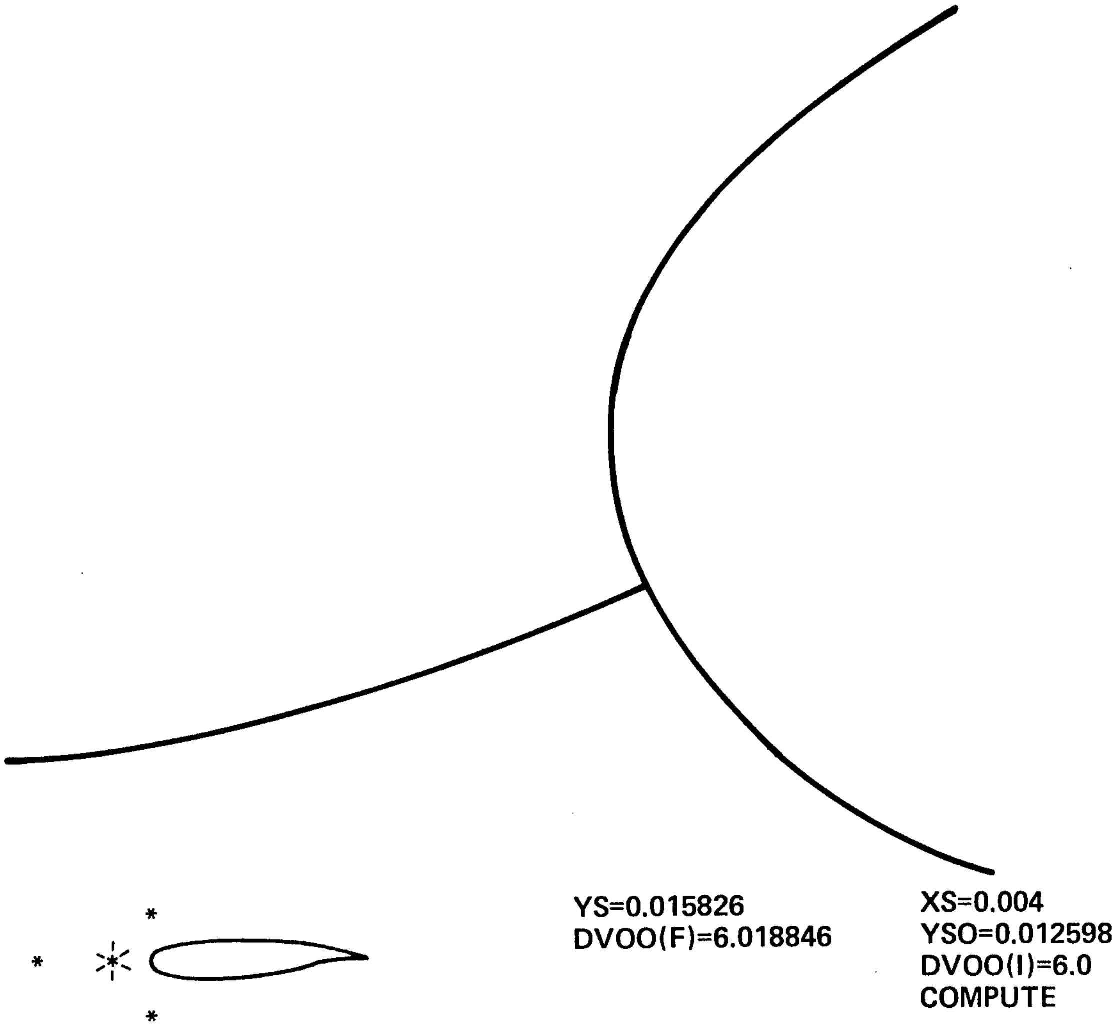


Figure 19 - Screen Display for Stagnation Solution

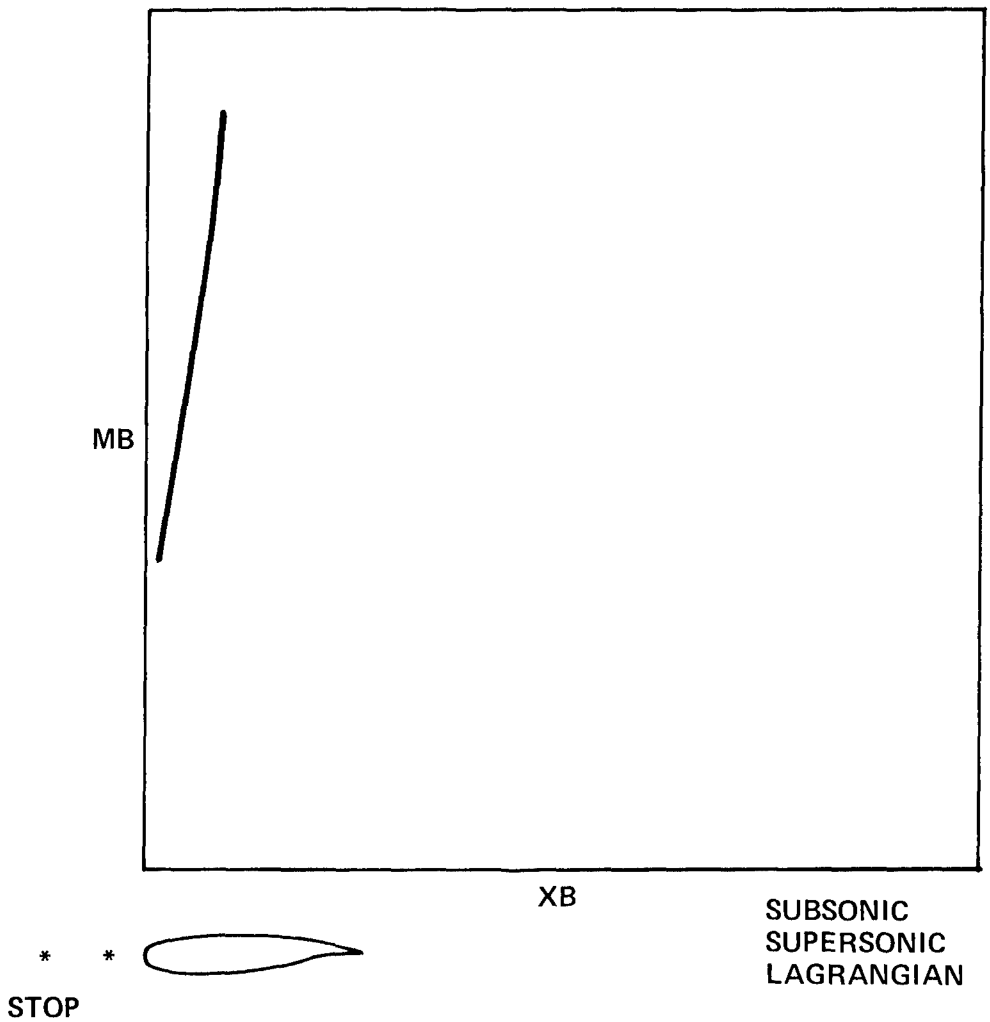


Figure 20 - Screen Display for Flow Criticality on Upper Surface

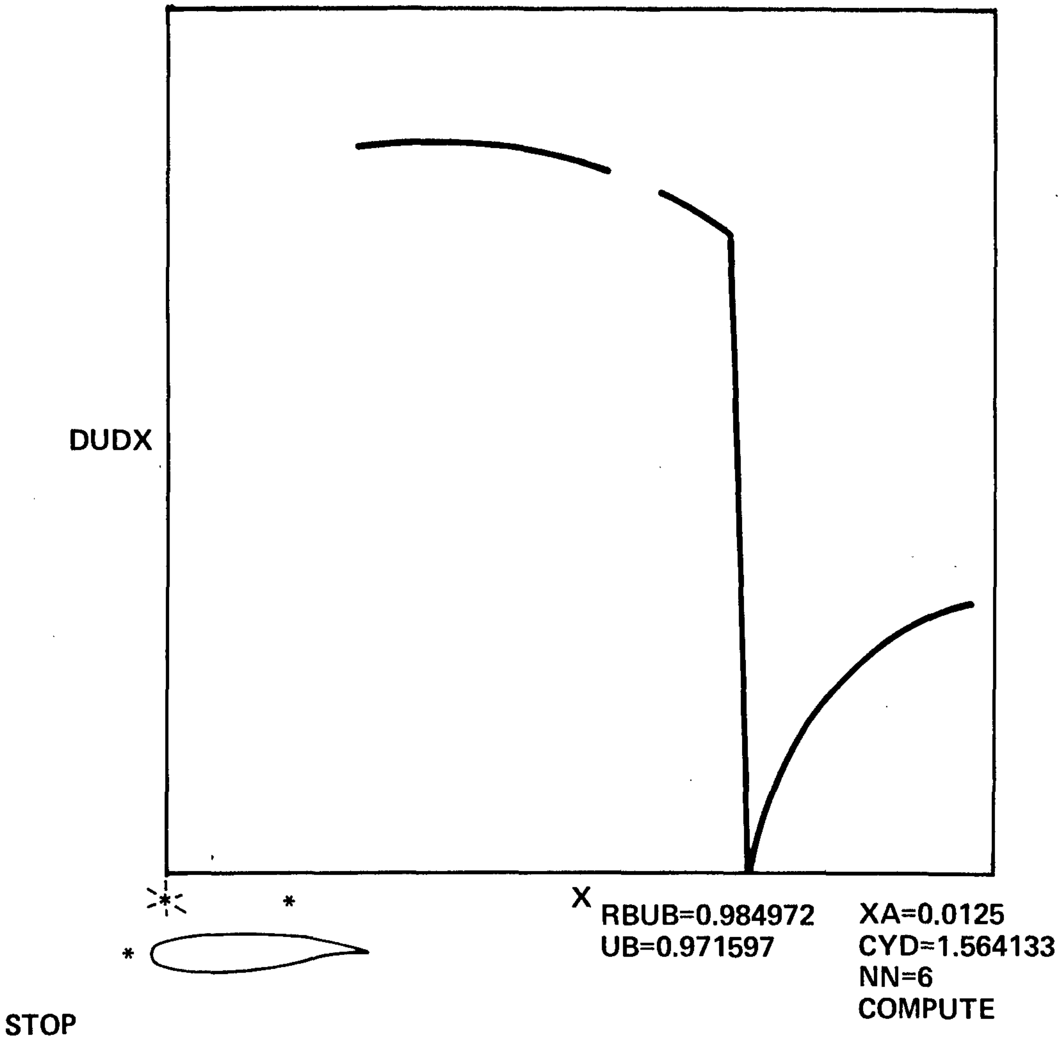


Figure 21 - Screen Display for Initial Solution — Upper Surface

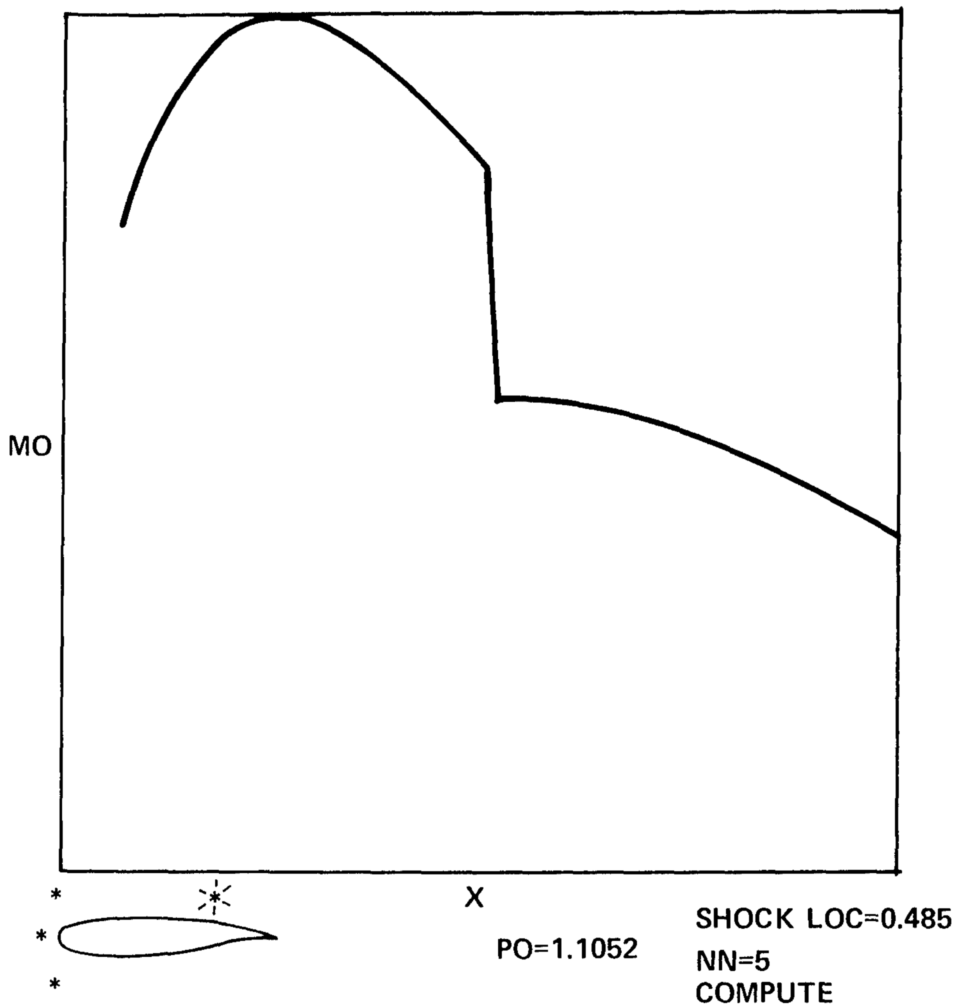


Figure 22 - Screen Display for Airfoil Solution — Upper Surface

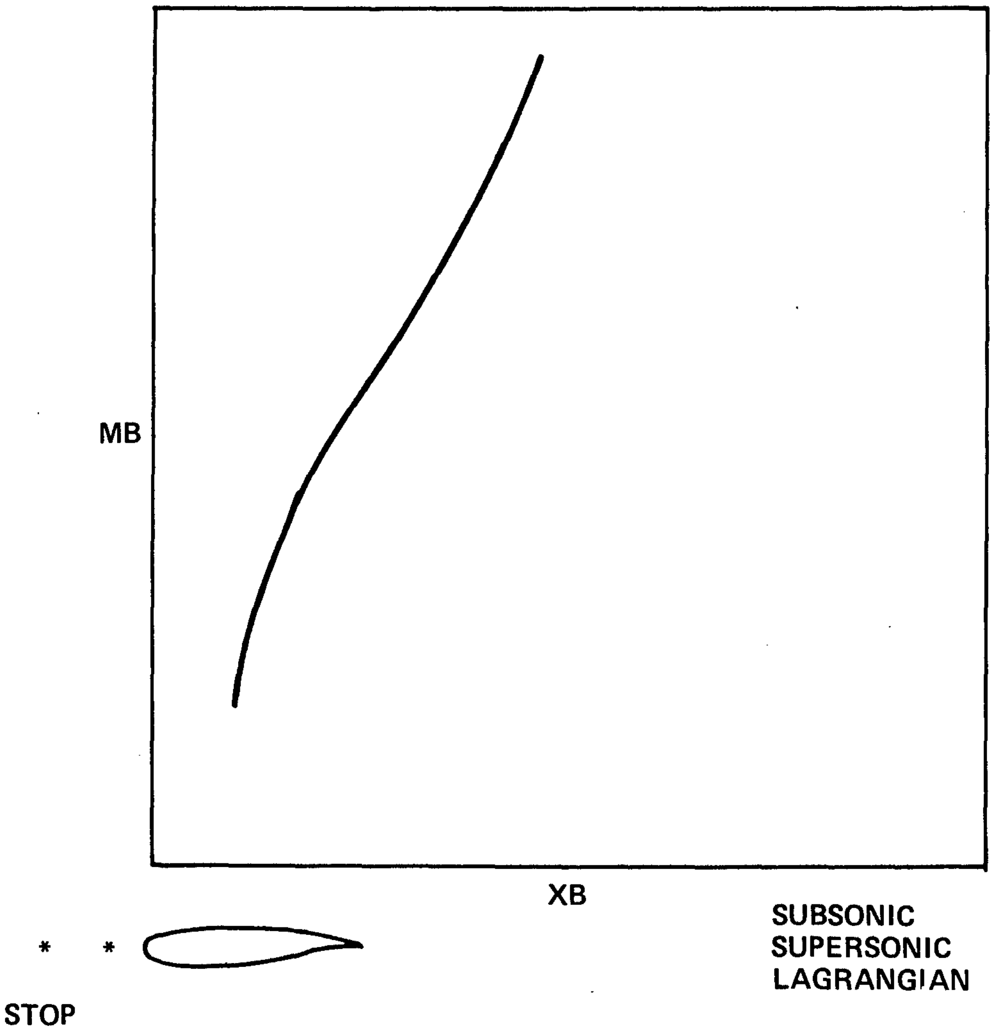


Figure 23 - Screen Display for Flow Criticality on Lower Surface

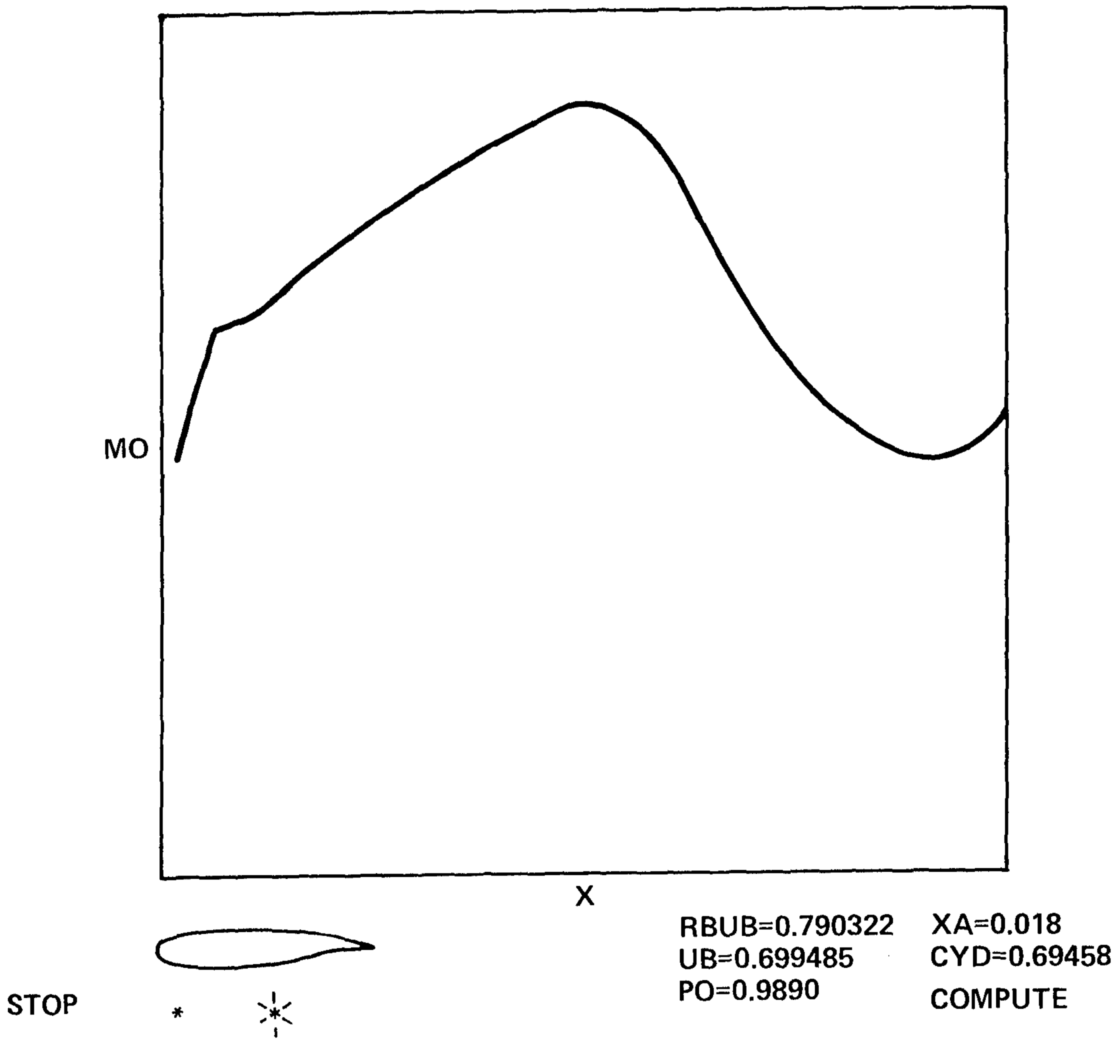


Figure 24 - Screen Display for Airfoil Solution — Lower Surface

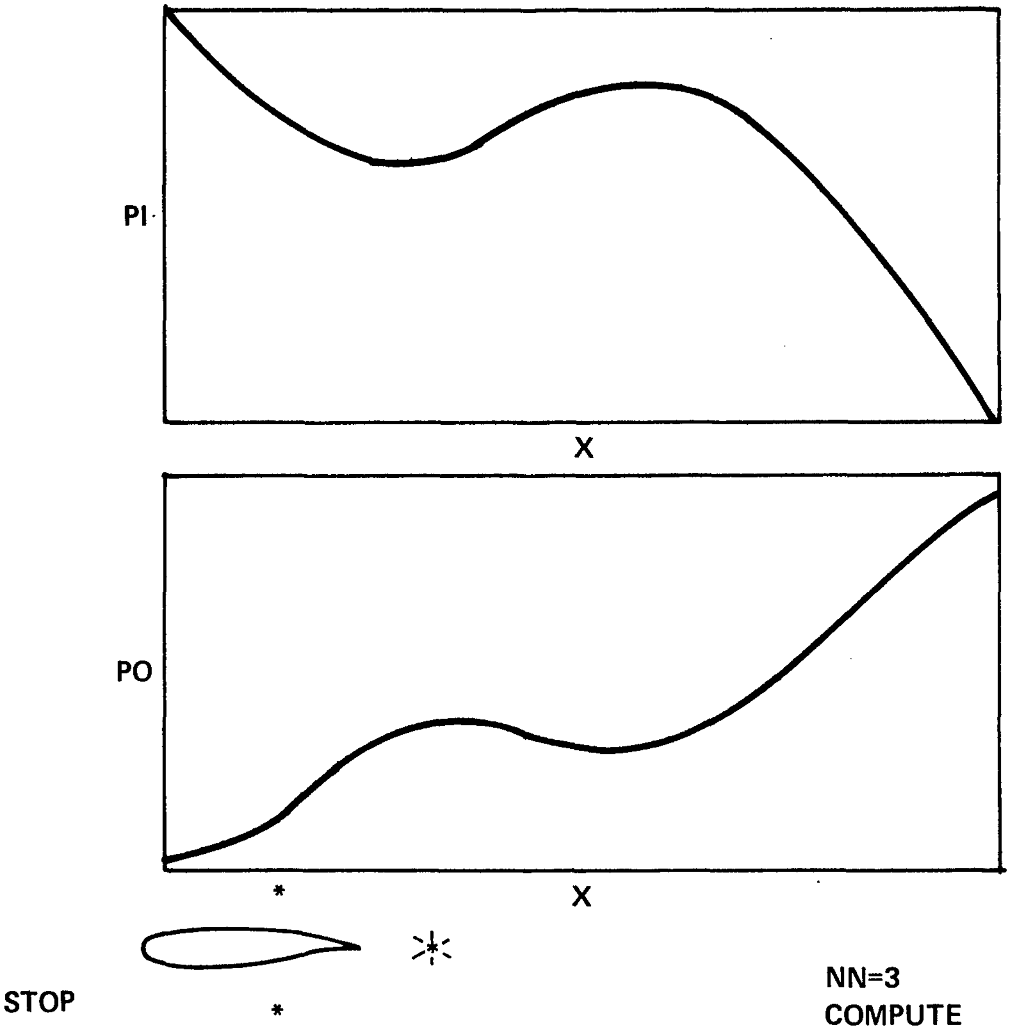


Figure 25 - Screen Display for Downstream Solution

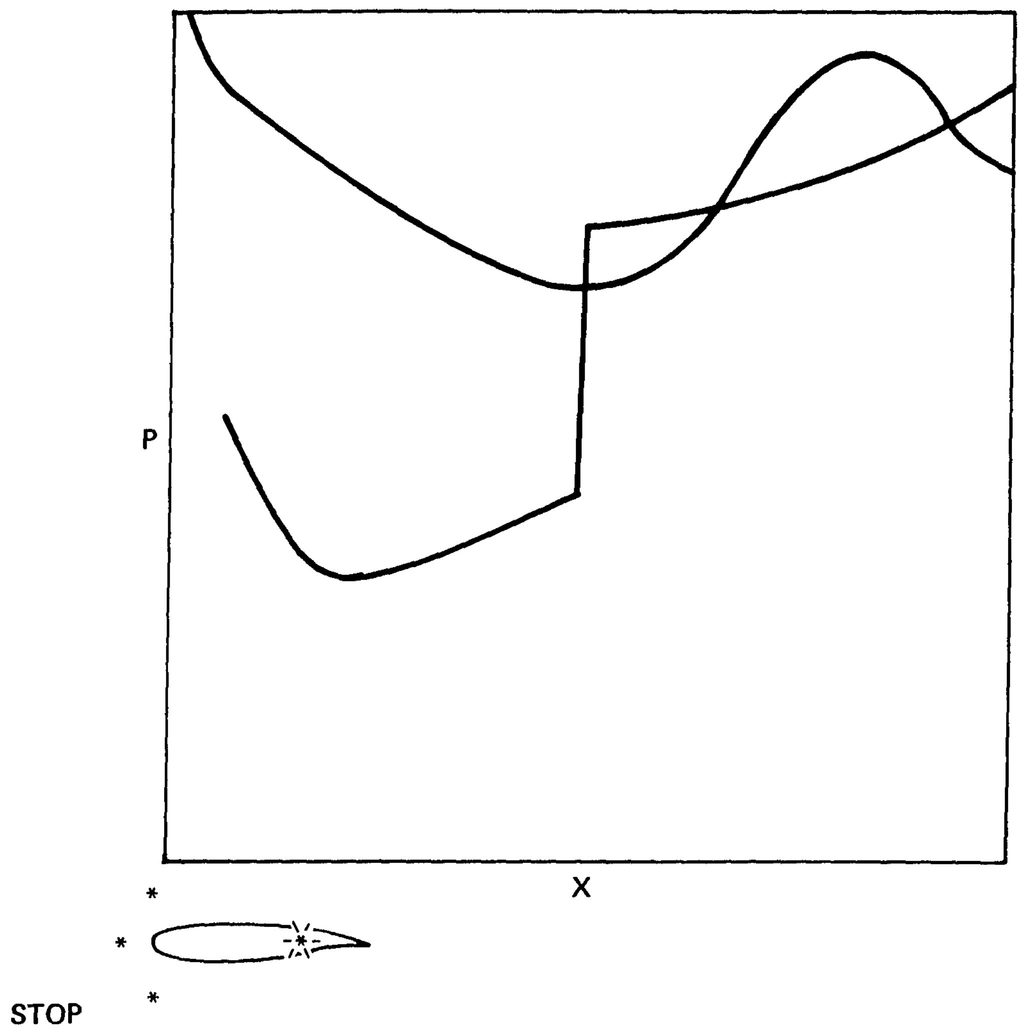


Figure 26 - Screen Display for Kutta Condition Check

APPENDIX A

AIRFOIL REPRESENTATION BY SPLINES

In the analysis of arbitrary airfoil shapes, very accurate first and second derivative information of the airfoil surface is needed to complete flow analysis. The use of a nonperiodic cubic spline in airfoil representations provides a function which has linear changes in the second derivative between points on the airfoil. In order to fit a cubic spline to the points on an airfoil, very accurate data input is necessary. Also necessary to generating an accurate cubic spline function are the initial and final slopes of the airfoil.

The determination of the coefficients m_j for a cubic spline function is given by the system of equations:[4]

$$\begin{bmatrix} 2 & \mu_o & 0 & \cdots & 0 & 0 & 0 \\ \lambda_1 & 2 & \mu_1 & \cdots & 0 & 0 & 0 \\ 0 & \lambda_2 & 2 & \cdots & 0 & 0 & 0 \\ & & & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & \mu_{N-2} & 0 \\ 0 & 0 & 0 & \cdots & \lambda_{N-1} & 2 & \mu_{N-1} \\ 0 & 0 & 0 & \cdots & 0 & \lambda_N & 2 \end{bmatrix} \begin{bmatrix} m_o \\ m_1 \\ m_2 \\ \\ m_{N-2} \\ m_{N-1} \\ m_N \end{bmatrix} = \begin{bmatrix} C_o \\ C_1 \\ C_2 \\ \\ C_{N-2} \\ C_{N-1} \\ C_N \end{bmatrix}$$

where C_j are the y-coordinates of the airfoil data points and m_j are the first derivatives at these data points. The coefficients $\lambda_j = h_1/(h_j+h_1)$ and $\mu_j = 1 - \lambda_j$. The variable h_j indicates the mesh spacing; here $h_j = x_j - x_{j-1}$ and x_j are the x-coordinates of the airfoil data points. The first derivatives at the beginning and end points (m_o and m_N) and the airfoil data points are assumed to be known.

Once the values of m_j have been determined, the cubic spline function can be expressed on (x_{j-1}, x_j) as

$$S(x) = m_{j-1}\frac{(x_j-x)^2(x-x_{j-1})}{h_j^2} - m_j\frac{(x-x_{j-1})^2(x_j-x)}{h_j^2}$$

$$+ y_{j-1}\frac{(x_j-x)^2[2(x-x_{j-1}) + h_j]}{h_j^3}$$

$$+ y_j\frac{(x-x_{j-1})^2[2(x_j-x) + h_j]}{h_j^3}$$

The first derivative can be expressed as

$$S'(x) = m_{j-1}\frac{(x_j-x)(2x_{j-1}+ x_j-3x)}{h_j^2} - m_j\frac{(x-x_{j-1})(2x_j+ x_{j-1}- 3x)}{h_j^2}$$

$$+ \frac{y_j - y_{j-1}}{h_j^3}\, 6(x_j - x)(x - x_{j-1})$$

The second derivative can be expressed as

$$S''(x) = -2m_{j-1}\frac{2x_j+ x_{j-1}- 3x}{h_j^2} - 2m_j\frac{2x_{j-1}+ x_j- 3x}{h_j^2}$$

$$+ 6\,\frac{y_j - y_{j-1}}{h_j^3}(x_j+ x_{j-1}- 2x)$$

The above represents a revised version of spline program as discussed by Tai.[2] More information on spline functions can be found in Ahlberg et al.[4]

---

4. Ahlberg, J. H. et al., "The Theory of Splines and Their Applications," Academic Press, New York (1967).

```
      SUBROUTINE SPLNFT(N, SLOPEI,SLOPEF,J)
      COMMON/PTARFL/XX(40,2) ,YY(40,2) ,AM(40,2) ,CA        ,S
      DIMENSION TRID(4,30)
      IF(N.NE.2) GO TO 20
      AM(1,J) =(YY(2,J)-YY(1,J))/(XX(2,J)-XX(1,J))
      AM(2,J) = AM(1,J)
      RETURN
   20 A = XX(2,J)-XX(1,J)
      B= XX(3,J)-XX(2,J)
      C = YY(2,J)-YY(1,J)
      D= YY(3,J)-YY(2,J)
      TRID(1,1) = 0.0
      TRID(2,1) =1.0
      TRID(3,1) = 0.0
      TRID(4,1) = SLOPEI
      TRID(1,N) = 0.0
      TRID(2,N) =1.0
      TRID(3,N) = 0.0
      TRID(4,N) = SLOPEF
      IFIN =  N-1
      DO 40 I=2,IFIN
      TRID(1,I)=A
      TRID(2,I) = 2.0 *(A+B)
      TRID(3,I) = B
      TRID(4,I)= 3.0*(A*D/B+B*C/A)
      IF(I-IFIN)35,50,35
   35 A=B
      B=XX(I+2,J)-XX(I+1,J)
      C=D
   40 D=YY(I+2,J)-YY(I+1,J)
   50 DO 55 I=1,IFIN
      TRID(1,I) = TRID(1,I)/TRID(2,I)
      TRID(4,I) = TRID(4,I)/TRID(2,I)
      TRID(2,I+1) = TRID(2,I+1) -TRID(3,I+1)*TRID(1,I)
   55 TRID(4,I+1)=TRID(4,I+1)-TRID(4,I)*TRID(3,I+1)
      TRID(4,N) = TRID(4,N)/TRID(2,N)
      AM(IFIN,J) = TRID(4,N-1)-TRID(1,N-1)*TRID(4,N)
      DO 60 I=2,IFIN
      NN = N-I
   60 AM(NN,J)=  TRID(4,NN) - TRID(1,NN)*AM(NN+1,J)
      AM(N,J) = TRID(4,N)
      RETURN
      END
```

APPENDIX B

DESCRIPTION OF SUBROUTINES

VARIABLES IN ACOM

The following variables refer to the dividing streamline, that is, the strip which proceeds from the upstream solution to the stagnation point and follows the upper and lower airfoil surfaces to the trailing edge of the body and from the trailing edge of the body to nine chord lengths downstream from the airfoil.

YO = y-coordinate normalized with respect to chord length

PO = pressure ratio P/P_∞

RO = density ratio ρ/ρ_∞

UO = velocity component in x, u/u_∞

VO = velocity component in y, v/v_∞

RMO = Mach number

DUO = velocity gradient $(du/dx)_o$

The following variables refer to the intermediate NN strips in flow integration:

Y(2,10) = y-coordinate normalized with respect to chord length

P(2,10) = pressure ratio P/P_∞

R(2,10) = density ratio ρ/ρ_∞

U(2,10) = velocity component in x, u/u_∞

V(2,10) = velocity component in y, v/v_∞

RM(2,10) = Mach number

DU(2,10) = velocity gradient $(du/dx)_o$

The first subscript references either upper or lower surface and the second subscript references a particular strip. For instance, Y(1,1) references the outermost strip on the upper surface and Y(2,1) references the outermost strip on the lower surface. If five strips are being used to perform the flow integration, Y(1,5) would reference the innermost strip on the upper surface. Suppose we chose in free-stream condition an outermost strip seven chord lengths away from the body; then Y(1,1) = 7.0. Intermediate strips are spaced half as far away from the dividing streamline as the previous strip. Hence,

Y(1,2) = 3.5

Y(1,3) = 1.75

Y(1,4) = 0.875

Y(1,5) = 0.4375

All of the flow variables are normalized by free-stream values.

VN = velocity component at airfoil surface normal to airfoil

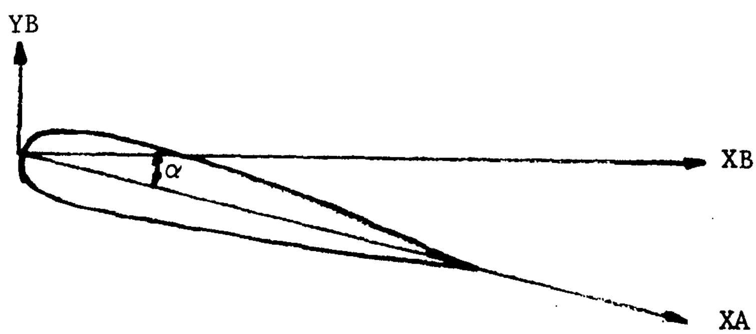
VS = velocity component at airfoil surface, tangential to airfoil

X = X-coordinate

XA = distance along airfoil chord

XB = distance along airfoil parallel to X axis

H = integration step size



VARIABLES IN COMMON/AINPUT/

AIN(1)	DVOOI	= estimated cross velocity gradient at stagnation point
AIN(2)	XS	= stagnation point on airfoil
AIN(3)	XAO	= initial point upper surface
AIN(4)	CYDU	= parameter indicating shape of final velocity profile of final upstream integration station for upper surface
AIN(5)	XAI	= initial point lower surface
AIN(6)	CYDL	= parameter indicating shape of final velocity profile of final upstream integration station for lower surface
AIN(7)	SL	= shock location

AIN(8)	XOO ≈ 4.0	= parameter used in calculating DVOO for UPSTRM
AIN(9)	RMT ≈ 7.0	= Mach number used in UPSTRM
AIN(10)	CDY ≈ 0.1	= upper limit for slope of stagnation streamline in UPSTRM
AIN(11)	YU ≈ 0.7	= location of outermost strip on upper surface
AIN(12)	YL ≈ 7.0	= location of outermost strip on lower surface
AIN(13)	CX ≈ 0.5	= X-coordinate for last integration step in SUBCRT1
AIN(14)	RMC ≈ 0.92	= upper limit for Mach number on innermost strip in SPRCRT1
AIN(15)	BETAD	= shock wave angle
AIN(16)	DELS	= entropy change through shock foot in SPRCRT2
AIN(17)	CDDQ ≈ 11.0	= upper limit for d^2q/dx^2 in SPRCRT2
AIN(18)	RKI ≈ 5.0	= value of RK for DWNSTRM
AIN(19)	XASPR	= X-coordinate for initial point in SPRCRT2
AIN(2)	DE	= distance from final station of upstream integration to airfoil surface
AIN(21)	YSO	= distance which stagnation streamline is perturbed by the airfoil
AIN(22)	YS	= y-coordinate of stagnation point
AIN(23)	CSI	= value of CS in DWNSTRM
AIN(24)	CZI	= value of CZ in DWNSTRM
NNI(1)		= number of strips used for the bulk of flow integration in UPSTRM
NNI(2)		= number of additional strips used near stagnation point in UPSTRM
NNI(3)		= number of strips used in UPRINIT
NNI(4)		= number of strips used in LWRINIT
NNI(5)		= number of strips used in SPRCRT2
NNI(6)		= number of strips used in DWNSTRM
NNI(7)		= dummy variable used in IOUPRIN and IOLWRIN
H(1)		= step size in UPSTRM
H(2)		= step size in SPRCRT1
H(3)		= step size in SUBCRT1
H(4)		= step size in SUBCRT2
H(5)		= step size in SPRCRT2
H(6)		= step size in DWNSTRM

VARIABLES IN COMMON/YUVSAV/

These variables contain the flow conditions output from one step and input to another. The flow conditions are y, the distance from the airfoil surface to this particular strip, u, the horizontal velocity component, and v, the vertical velocity component.

For subroutine UPSTRM, the output variables are stored in arrays YI, UI, and VI, and the number of strips is stored in NNI. These variables are input to subroutines UPRCRIT, UPRINIT, LWRCRIT, and LWRINIT.

For subroutine SPRCRT1, the output variables are stored in arrays YSPR, USPR, and VSPR, and the number of strips is stored in NNSPR. These variables are input to subroutine SPRCRT2.

For subroutine SPRCRT2 or SUBCRT2, the output variables are stored in arrays YU, UU, VU, and YL, UL, and VL, and the number of strips is stored in NNDWN. The flow conditions for the dividing streamline are stored in arrays YO, UO, and VO. These variables are input to subroutine DWNSTRM.

VARIABLES IN BLANK COMMON

$C = 1 + 5/M^2$

$CK = 5/7M^2$

$RS = (1/7CK + 1)^{2.5}$

FM = M = Mach number

ALPHA = α = angle of attack

OUTPUT SUBROUTINES

No references are made to read and write units in the subroutines which actually perform the flow integration processes. The output variables are stored in arrays and are output on the line printer in subroutines beginning with the letters IO. The output subroutines and the corresponding flow integration subroutine are:

IOUPSTM	UPSTRM
IOSTGNA	STAGNA
IOUPRCT	UPRCRIT
IOUPRIN	UPRINIT,SPRCRT1
IOSPCT2	SPRCRT2
IOLWRCT	LWRCRIT
IOLWRIN	LWRINIT,SUBCRT1,SUBCRT2
IODNSTM	DWNSTRM

FLOW INTEGRATION SUBROUTINES

Subroutine UPSTRM

This subroutine performs the upstream integration from free-stream conditions to the stagnation point on the airfoil. The primary outputs of the subroutine are given in COMMON/OUTCOM.

AXA = x stations along stagnation streamline
ARMO = Mach number along stagnation streamline
AY = y stations at final integration station
ARM = Mach numbers at final integration station

Other important variables are

CSO = Mach number
CS1 = Mach number at which more strips are added to the flow solution
DYO = local slope of the stagnation streamline

Integration proceeds from the free-stream conditions to a point where the Mach number on the stagnation streamline is considerably lessened. This part of the integration uses subroutine DIST which

includes the effects of the cross velocity gradient DVOOI. When the Mach number RMO reaches a particular value of RMT, the flow integration uses subroutines STMR and LUMR depending on the slope of the stagnation streamline, DYO. The value of RMO should decrease until it reaches a value of CSO; at this point, flow values are stored for future use. If this is the first time that flow variables are stored, more strips are added to the flow integration. The flow integration now includes a total of NN + NA strips and the stagnation streamline. The parameter CSO is decreased by a value of 0.05, and flow integration continues to the point where RMO reaches this value. The flow values are stored at this point, and the process is repeated until flow values are stored at four points. By using a Lagrangian, values of RMO are extrapolated to the point where RMO = 0, the point where the stagnation streamline meets the airfoil. The value of DE, the distance from the last computed station of upstream integration to the stagnation point on the airfoil, can now be computed. Upstream integration is now complete.

The final section of this subroutine computes airfoil coordinates and 11 points along the stagnation streamline. This information is input to subroutine STAGNA to calculate the stagnation streamline geometry.

Subroutine STAGNA

This subroutine computes the cross velocity gradient at the point XS and the stagnation streamline geometry corresponding to this stagnation point. The primary outputs of this subroutine are given in COMMON/ECOM/ as:

XOU = x-coordinates of airfoil nose

YOU = y-coordinates of airfoil nose

XAF = x-coordinates of stagnation streamline

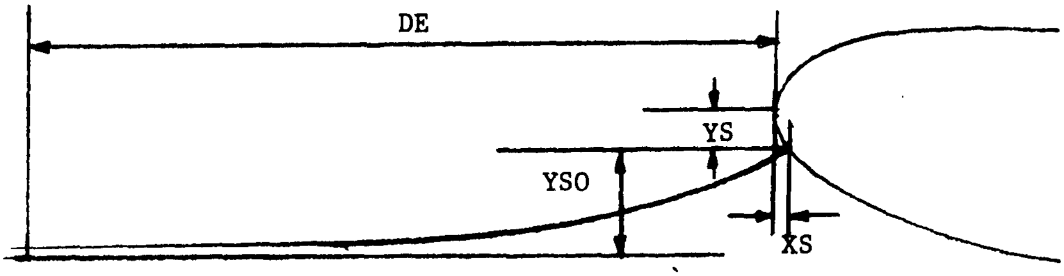
YAF = y-coordinates of stagnation streamline

Other important variables in this subroutine are:

DE = distance from final station of upstream integration to airfoil surface

YSO = distance which stagnation streamline is perturbed by the airfoil

YS = y-coordinate of stagnation point



For a given stagnation point XS, the radius of curvature RA at this point on the airfoil is calculated. The cross velocity gradient DVOOF can then be calculated. The airfoil coordinates and stagnation streamline are then converted to one Cartesian frame of reference.

Subroutine UPRCRIT

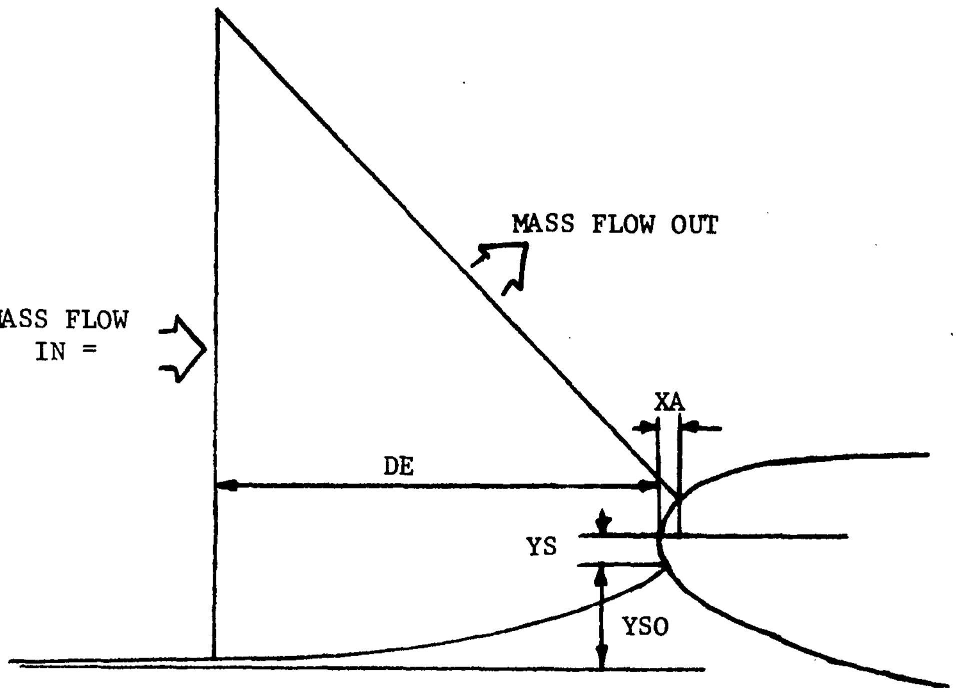
This subroutine determines flow criticality on the upper surface by computing the Mach number at various points along the surface. The primary outputs of this subroutine are given in COMMON/OUTCOM/ as:

AXB = x station on airfoil surface

ARMB = Mach number calculated at airfoil surface

For a given initial point XA, a perpendicular is drawn to the airfoil surface which intersects the final station of upstream integration. Depending on where the perpendicular intersects this station, the mass flow into the control volume is calculated. By using the Newton-Raphson method, the flow out of the control volume is calculated which matches the input flow, and the Mach number at this point is calculated.

The sketch illustrates the control volume and the corresponding geometry.



Subroutine UPRINIT(ICRIT)

This subroutine determine the initial flow conditions on the upper surface for a particular initial point. The solution method is the same as UPRCRIT, but the solution is performed for one point instead of a series of points. Important output variables are given in COMMON/RBUBCM/ as:

RBUB = mass flux at initial point

UBINIT = velocity at initial point

The calculation of RBUB, the mass flux at the surface of the airfoil, includes the term CYD. Increasing the value of CYD decreases the value of RBUB. In some cases the value of UBINIT cannot be calculated because the value of RBUB is too large or too small, but there is a wide range of values of CYD for which a solution exists.

Subroutine SPRCRT1(J)

This subroutine performs the initial integration of flow conditions for supercritical flow. The parameter J indicates airfoil surface. (J = 1 for upper surface, J = 2 for lower surface.) Important output variables are given in COMMON/OUTCOM/ as:

XBO = x stations at airfoil surface

RMBO = Mach number along stagnation streamline

DUBO = velocity gradients along stagnation streamline

The subroutine consists of three main steps:

1. Integration in subsonic region and storage of data during integration to extrapolate through the sonic point.

2. Extrapolation through the sonic point.

3. Integration in supersonic region and extrapolation of data to final station.

Flow integration in this subroutine as well as SUBCRT1 utilizes subroutine INBO to calculate the flow properties at the surface of the airfoil. All other subroutines calculate flow properties along the airfoil surface in subroutine INAS.

A typical flow integration step in subroutine SPRCRT1 has the form

```
   CALL OUNS(1,J)
   NN1 = NN-1
   DO 10 N = 1, NN1
10 CALL INAS(1,J,N,1)
   CALL INBO(NN,J)
```

During integration in the subsonic region, checks are made at each integration step on the surface velocity gradient DUB and Mach number RMB. If DUB is less than 5.0 or RMB suddenly becomes greater than 1.0, the trial is aborted. For purposes of extrapolating data through the sonic point, data are saved at points where RMB = 0.9, 0.92, 0.94 and 0.96.

Once data at those four points have been saved, an attempt is made to extrapolate through the sonic point by using a Lagrangian function. The value of XA is incremented until a value for RMB greater than 1.03

is produced. Once this is done, the flow properties at this station are calculated, and integration proceeds to the supersonic region.

During integration in the supersonic region, checks are made at each integration step on the values of DUB and RMB. If DUB is greater than 60.0 or RMB is less than 1.0, the trial is aborted. If the value of the velocity gradient and Mach number at the innermost strip, DU(J,NN) and RM(J,NN), respectively, are greater than specified values, the number of integration strips is reduced by one. Because of different coordinate systems used, a very small gap in the flow field exists between the output station of SPRCRT1(J) and the input station of SPRCRT2(J). Flow properties are extrapolated in this gap by using a Lagrangian function.

Subroutine SPRCRT2(J)

This subroutine performs integration for supercritical flow for the bulk of the airfoil surface. The output variables located in COMMON/OUTCOM/ are:

AXA = integration station XA

ADU = velocity gradient (du/dx) at innermost strip NN

DDQO = d/dx(dq/dx) spatial rate of change of dynamic pressure at innermost strip NN

A typical flow integration step in subroutine SPRCRT2 has the form

```
      CALL OUNS(1,J)
      DO 10 N = 1, NN
   10 CALL INAS(1,J,N,NN)
```

Flow integration in the upper surface is accomplished in three steps: subcritical, supercritical, and subcritical flow integrations. Subcritical flow is calculated from the leading edge to the sonic line; supercritical from sonic line to shock location SL and finally subcritical from SL to the trailing edge.

During flow integration in the supersonic region, checks are made on the values of DUO and RMO. If DUO is greater than 100 or RMO is less than 1.0, the trial is aborted. If the value of DDQ is greater than the input value of CDDQ, the number of flow integration strips is decreased by one.

Flow integration proceeds to the point where X is greater than SL, and the Rankine-Hugoniot relations are applied there across the designated shock location. If the value of DELS is greater than 0.0, allowances are made for an entropy change through the shock location.

New flow variables are computed at the shock location SL, and flow integration proceeds to the trailing edge. Checks are made on DUO and RMO to ensure that the flow remains subcritical throughout the integration.

Subroutine LWRCRIT

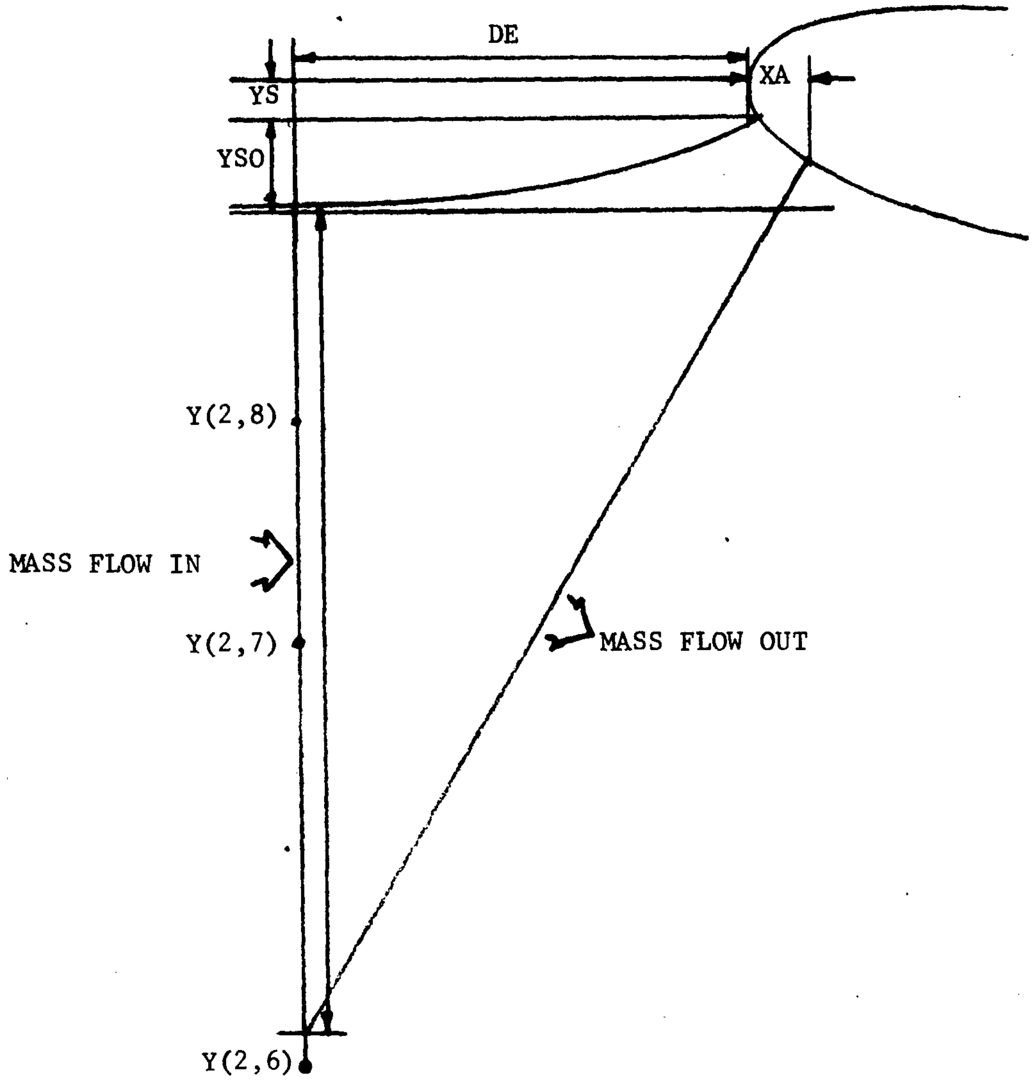
This subroutine determines flow criticality on the lower surface by computing Mach number at various points along the surface. Important output variables given in COMMON/OUTCOM are:

AXB = X stations for Mach number calculations

ARMB = Mach number calculated at airfoil surface

For a given initial point XA, a perpendicular is drawn to the airfoil surface which intersects the final station of upstream integration. Depending on where the perpendicular intersects this station, the mass flow into the control volume is calculated. By using the Newton-Raphson method, the mass flow out of the control volume is calculated which matches the input flow, and the Mach number is calculated.

An illustration of the control volume and the corresponding geometry is shown below; the perpendicular intersects the final station between Y(2,6) and Y(2,7).



Subroutine LWRINIT(ICRIT)

This subroutine determines the initial flow conditions on the lower surface for a particular initial point. The solution method is the same as LWRCRIT, but the solution is performed for one point instead of a series of points. Important output variables given in COMMON/RBUBCM/ are:

RBUB $= \rho_b V_b$ at initial point

UBINIT = velocity at given initial point

The initial flow conditions for lower surface flow are computed in the same manner as subroutine LWRCRIT, and computation proceeds according to the value of ICRIT.

Subroutine SUBCRT1(J)

This subroutine performs the initial integration of flow conditions for subcritical flow. Important output variables are given in COMMON/ OUTCOM/ as:

XBO = integration station

RMBO = Mach number along stagnation streamline

DUBO = velocity gradient along stagnation streamline

A typical flow integration step in subroutine SUBCRT1 has the form

```
      CALL OUNS(1,J)
      NN1 = NN-1
      DO 10 N = 1, NN1
   10 CALL INAS(1,J,N,1)
      CALL INBO(NN,J)
```

where NN is the number of strips available for integration.

During flow integration, checks are made on the values of RMO and DUB. If RMO is greater than 1.0, the assumption of subcritical flow is invalid and the trial is aborted. The trial is also aborted if the velocity gradient DUB becomes negative near the leading edge of the airfoil.

Data are saved at four stations for use in extrapolation to the output flow conditions at station XB which is fairly close to CXI. The extrapolation is needed to proceed with the integration in subroutine SUBCRT2.

Subroutine SUBCRT2(J)

This subroutine performs integration for subcritical flow for the bulk of the airfoil surface. The output variables located in COMMON/ OUTCOM/ are:

AXA = integration station

ARMO = Mach number at surface

ADU = velocity gradient (du/dx) at innermost strip NN

A typical flow integration step in subroutine SUBCRT2 has the form

```
    CALL OUNS(1,J)
    DO 10 N = 1, NN
 10 CALL INAS(1,J,N,NN)
```

During flow integration checks are made on the values of RMO and ADU. If RMO does not lie between 0.4 and 1.0, the trial is aborted and if the absolute value of the velocity gradient $(du/dx)_{NN}$ becomes greater than 2.0, the innermost strip is dropped. At a value of $X = CX = 0.5$, another strip is added between the innermost strip and the airfoil surface, and flow integration continues to the trailing edge of the airfoil, $X = CX = 1.0$.

Subroutine DWNSTRM

This subroutine performs flow integration downstream in the airfoil. The output variables located in COMMON/OUTCOM/ are:

AX = integration station

APO = pressure ratios along dividing streamline

AP1 = pressure ratios on NN strip

A typical flow integration step in subroutine DWNSTRM has the form

```
    CALL OUNS(1,1)
    CALL INAS(1,1,NN,NN)
```

where NN is the number of the strip used to integrate the intermediate strip.

Subroutine INVELOC(L,J)

This subroutine calculates the velocity component V(J,NN) of the innermost strip. If L = 2, this velocity component is computed according to a Lagrangian function; if L = 3, this velocity component is computed according to a parabolic function.

Subroutine ARFL(XA,XB,YB,DYB,DDYB,J)

This subroutine determines the y-coordinate and its first and second derivatives at a point on the airfoil. The arguments of this subroutine are:

XA = airfoil x-coordinate

The following values are calculated at the given angle of attack:

XB = x-coordinate

YB = y-coordinate

DYB = first derivative

DDYB = second derivative

J = 1 for upper surface and 2 for lower surface.

These values are determined according to the equations in Appendix A.

Subroutine DIST(M,I,N,DY1,DVS,DV1)

This subroutine performs a flow integration step on the divising streamline in the upstream solution. The arguments of this subroutine are:

M = ± 1, indicating direction of integration

I = 1 for upper surface and 2 for lower surface

N = number of innermost strip

DY1 = slope of dividing streamline

DVS = increment in velocity of dividing streamline

DV1 = increment in velocity of vertical component of dividing streamline

This subroutine includes the effect of the cross velocity gradient DVOO in determining the flow conditions far upstream from the airfoil. As the flow integration approaches the airfoil, the Mach number becomes too small; the flow integration must be completed by subroutines STMR and LUMR.

Subroutine STMR(N,T,DY,DVS)

This subroutine performs a flow integration step on the dividing streamline in the upstream solution. The arguments of this subroutine are:

N = number of innermost strip

T = angle of dividing streamline with respect to the horizontal; $T = SIN^{-1} DY/\sqrt{1 + DY^2}$

DY = slope of dividing streamline

DVS = increment in velocity of dividing streamline

This subroutine neglects the effects of changes in the vertical component of the dividing streamline in computing the flow integration. This is valid if DY > 0.1.

Subroutine LUMR(M,I,N,DY1,DVS,DV1)

This subroutine performs a flow integration step on the dividing streamline in the upstream solution. The arguments of this subroutine are:

M = ± 1, indicating direction of integration

I = 1 for upper surface and 2 for lower surface

N = number of innermost strip

DY1 = slope of dividing streamline

DVS = increment in velocity of dividing streamline

DV1 = increment in velocity of vertical component of dividing streamline

This subroutine computes the flow variables in the vicinity of the airfoil when the value of DY1 is less than 0.1.

Subroutine OUNS(M,I)

This subroutine performs a flow integration step on the next to the outermost streamline. The arguments of the subroutine are:

M = ± 1, indicating direction of integration

I = 1 for upper surface and 2 for lower surface

The flow on the outermost strip is assumed to be undisturbed and this integration is performed on the next strip. The remaining strips in the flow field are computed in subroutine INAS.

The gradients calculated in this subroutine are:

DU(I,2) = du/dx, velocity in x-direction

DV1 = dv/dx, velocity in y-direction

DY1 = dy/dx, slope of streamline

as output of the Runge-Kutta integration process. Gradients calculated as input to the next inner strip are:

DRHU(I) = $d/dx(\rho u)$

DPRU(I) = $d/dx(KP + \rho u^2)$

DRUV(I) = $d/dx(\rho uv)$

Subroutine INAS(M,I,N,IJ)

This subroutine performs a flow integration step on the Nth strip. The arguments of this subroutine are:

M = 1, indicating direction of integration

I = 1 for upper surface and 2 for lower surface

N = strip number

IJ = NN for subroutines SUBCRT2, SPRCRT2, and DWNSTRM

IJ = 1 for all other subroutines

This subroutine performs the bulk of flow integration from the third to the NNth strip. For subroutines SUBCRT2, SPRCRT2, and DWNSTRM, it also performs integration on the dividing streamline. This is accomplished by the use of the parameter ISKIP, which indicates for which subroutine the solution is being computed.

ISKIP	SUBROUTINE
1	SUBCRT2
2	SPRCRT2
3	DWNSTRM

The value of VO, the vertical component of the stagnation streamline, is computed differently in these three subroutines.

vo = uo * dy/dx for SUBCRT2

vo = vo + h (dv_o/dx) for SPRCRT2 where dv_o/dx is computed according to MIR

vo = $vo_{T.E.}$ exp[(1-x)RK] for DWNSTRM

The gradients calculated in this subroutine are the same as those for subroutine OUNS.

Subroutine INBO(N,I)

This subroutine performs a flow integration along the dividing streamlines for subroutines SUBCRT1 and SPRCRT1. The arguments of this subroutine are:

N = number of innermost strip

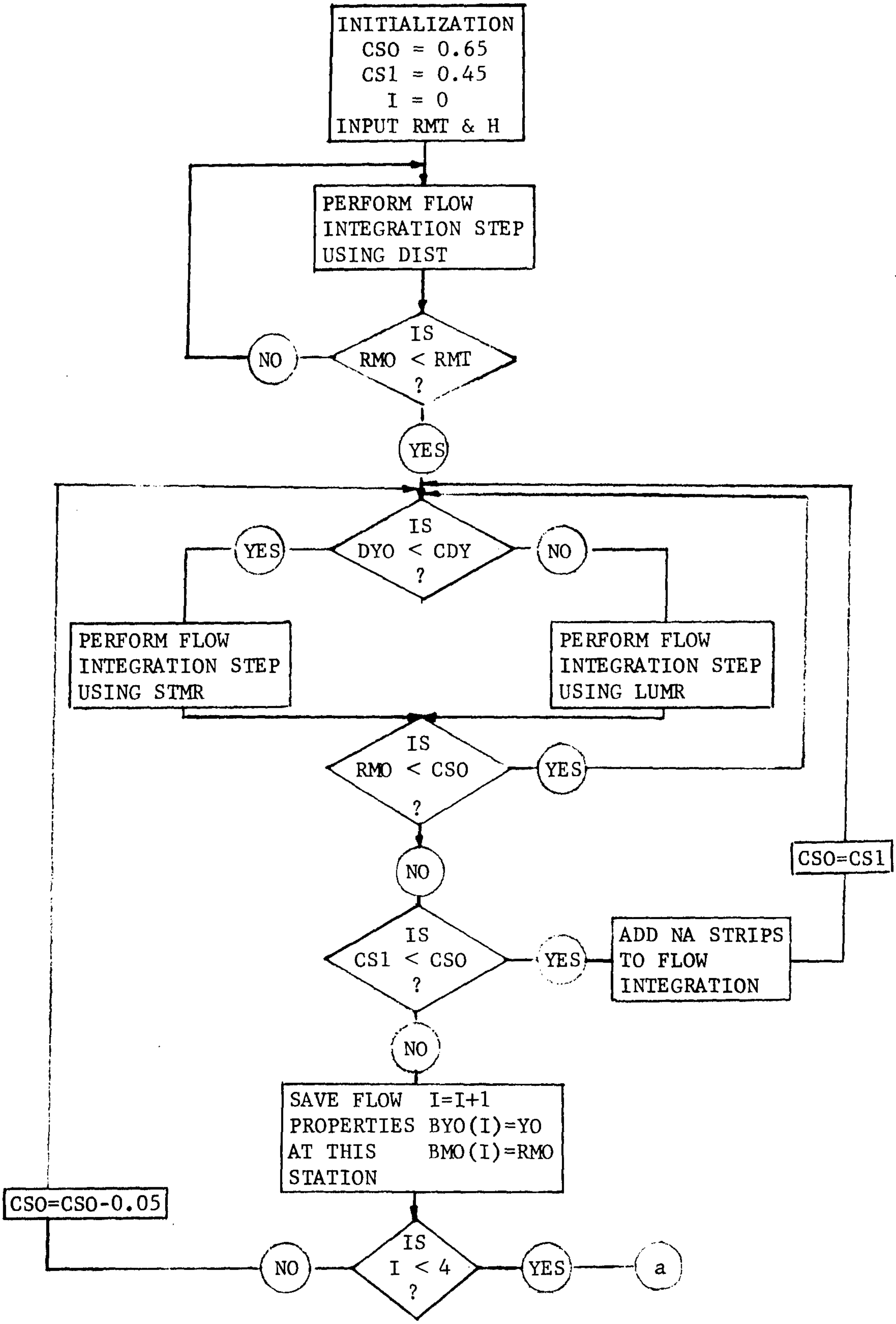
I = 1 for upper surface and 2 for lower surface

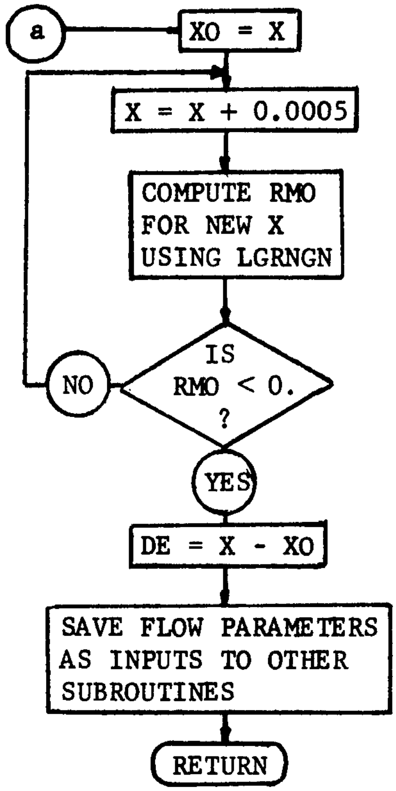
The calculation of a strip between the innermost strip and the airfoil surface is also accomplished here. The parameters YO, UO, VO, RO, and PO refer to this strip and the parameter YB, UB, RB, and PB refer to the values at the airfoil surface along the dividing streamline.

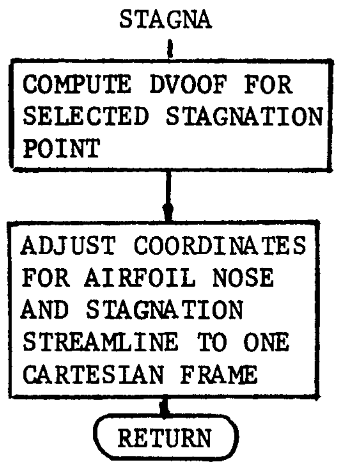
APPENDIX C

SUBROUTINE FLOW CHARTS

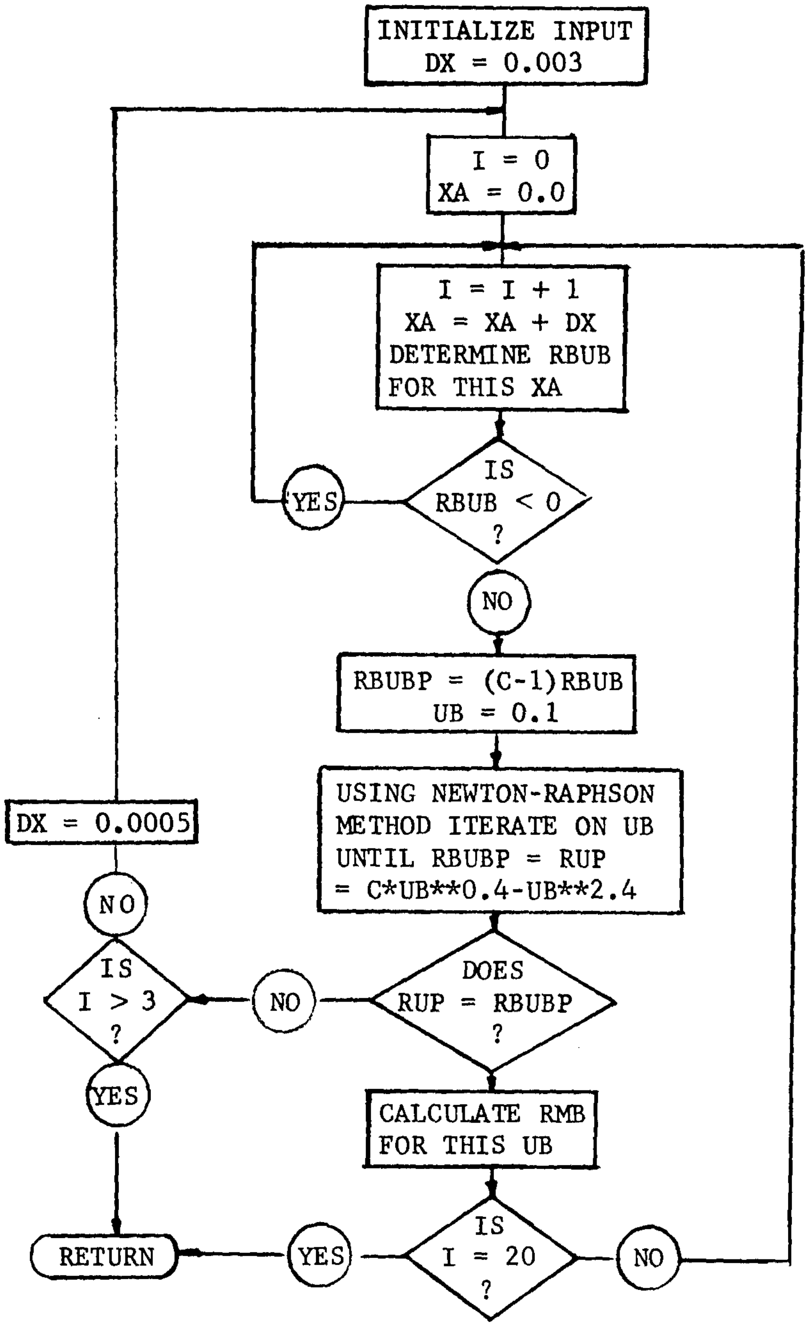
UPSTRM

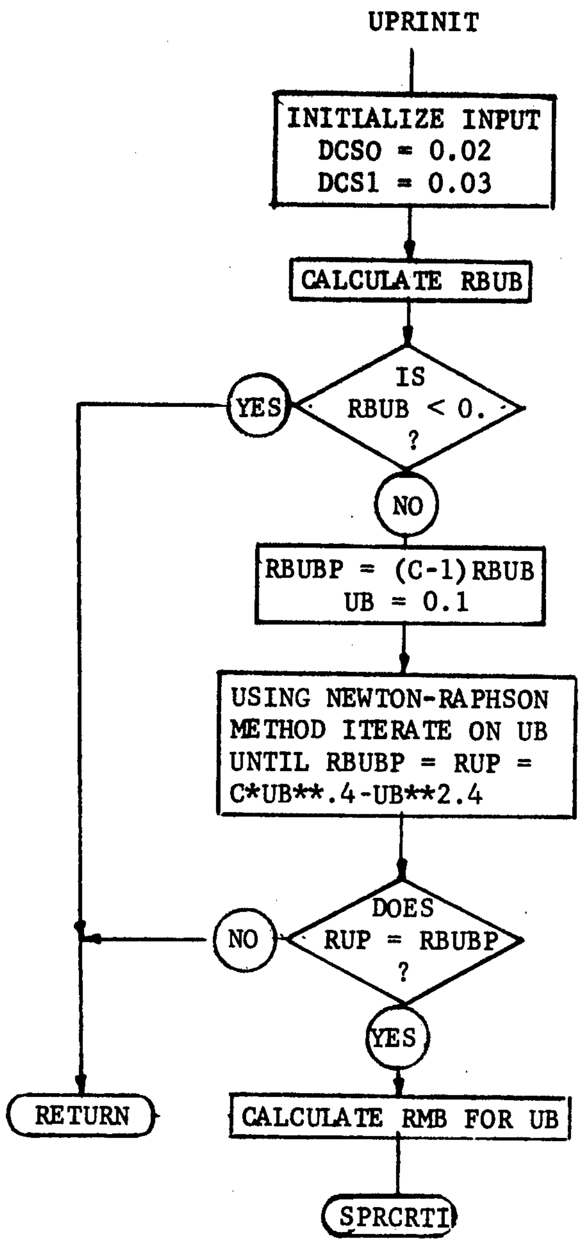



a
XO = X
X = X + 0.0005
COMPUTE RMO FOR NEW X USING LGRNGN
IS RMO < 0. ?
NO
YES
DE = X - XO
SAVE FLOW PARAMETERS AS INPUTS TO OTHER SUBROUTINES
RETURN

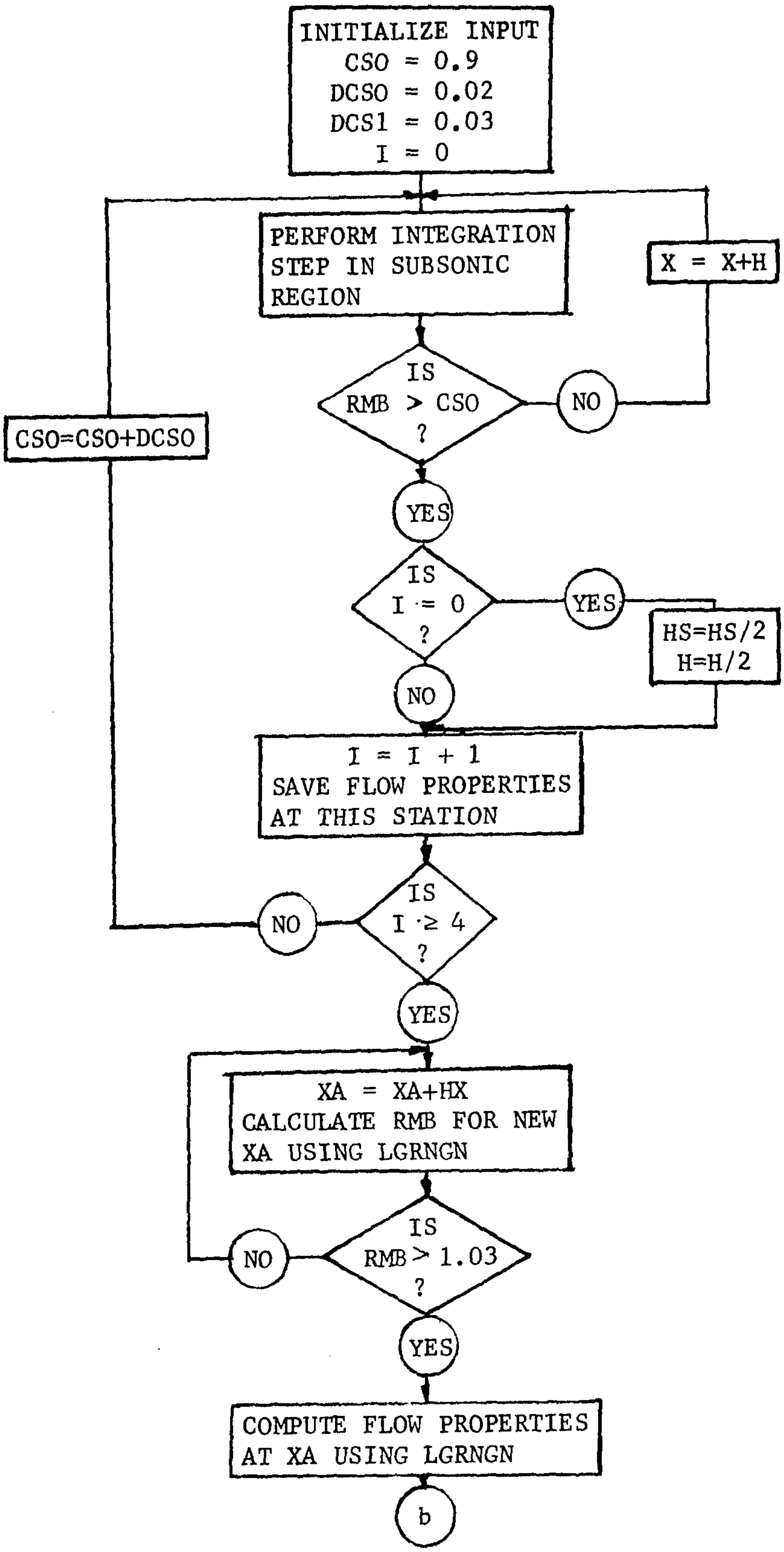

STAGNA
COMPUTE DVOOF FOR SELECTED STAGNATION POINT
ADJUST COORDINATES FOR AIRFOIL NOSE AND STAGNATION STREAMLINE TO ONE CARTESIAN FRAME
RETURN

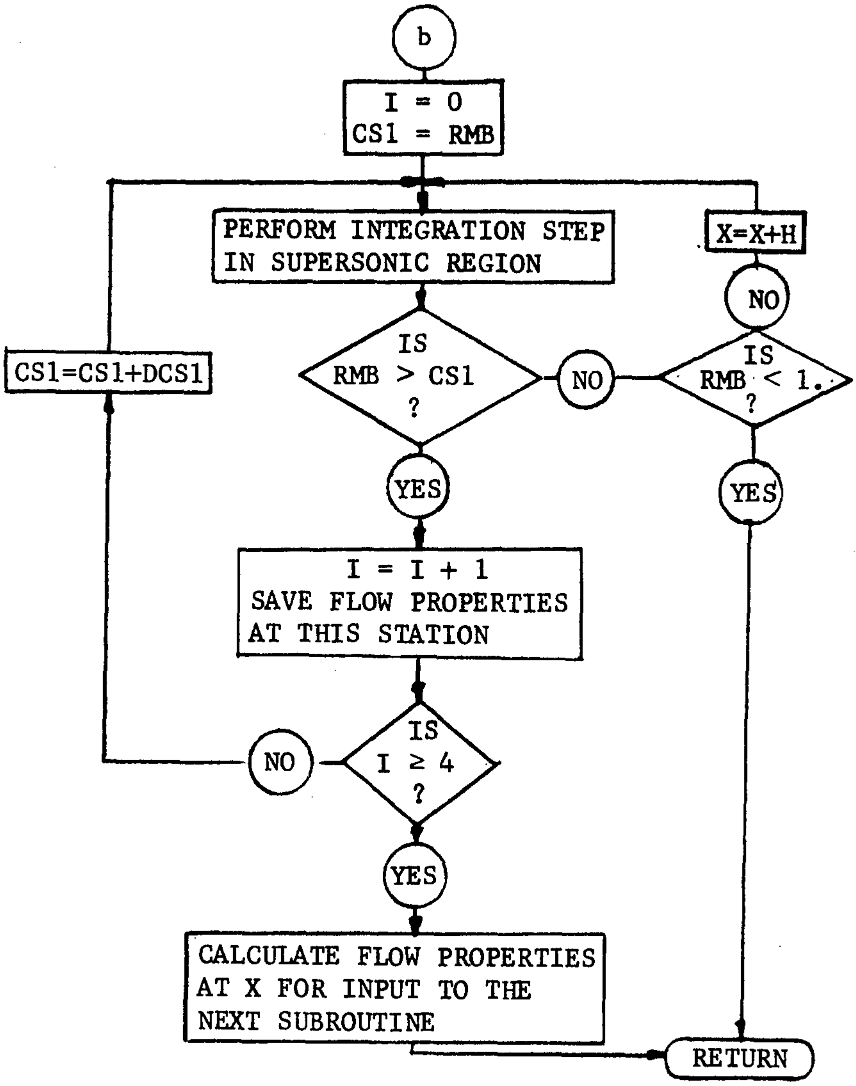
UPRCRIT

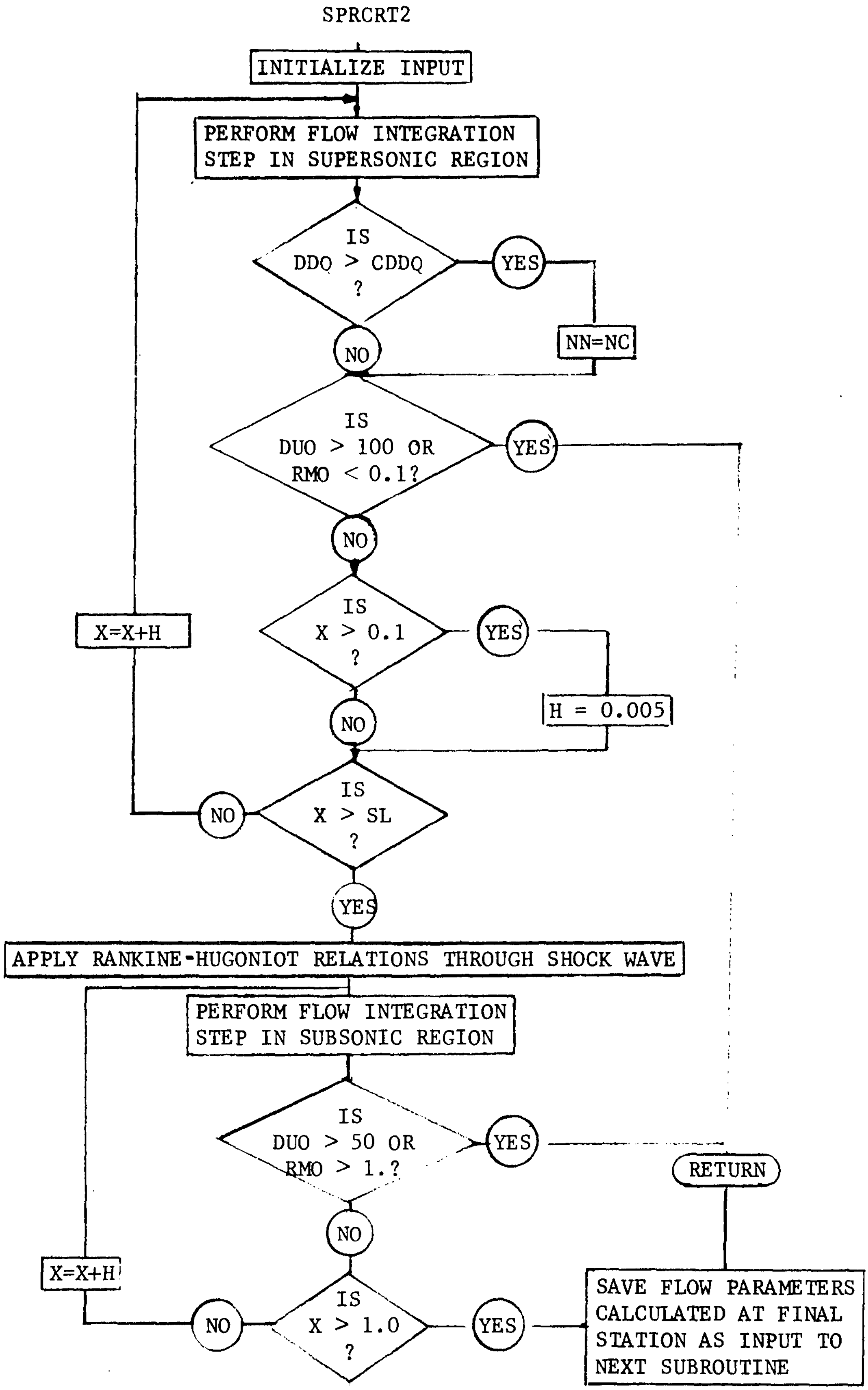

INITIALIZE INPUT
DX = 0.003
I = 0
XA = 0.0
I = I + 1
XA = XA + DX
DETERMINE RBUB
FOR THIS XA
IS
RBUB < 0
?
YES
NO
RBUBP = (C-1)RBUB
UB = 0.1
USING NEWTON-RAPHSON
METHOD ITERATE ON UB
UNTIL RBUBP = RUP
= C*UB**0.4-UB**2.4
DOES
RUP = RBUBP
?
NO
IS
I > 3
?
NO
DX = 0.0005
YES
CALCULATE RMB
FOR THIS UB
IS
I = 20
?
YES
NO
RETURN

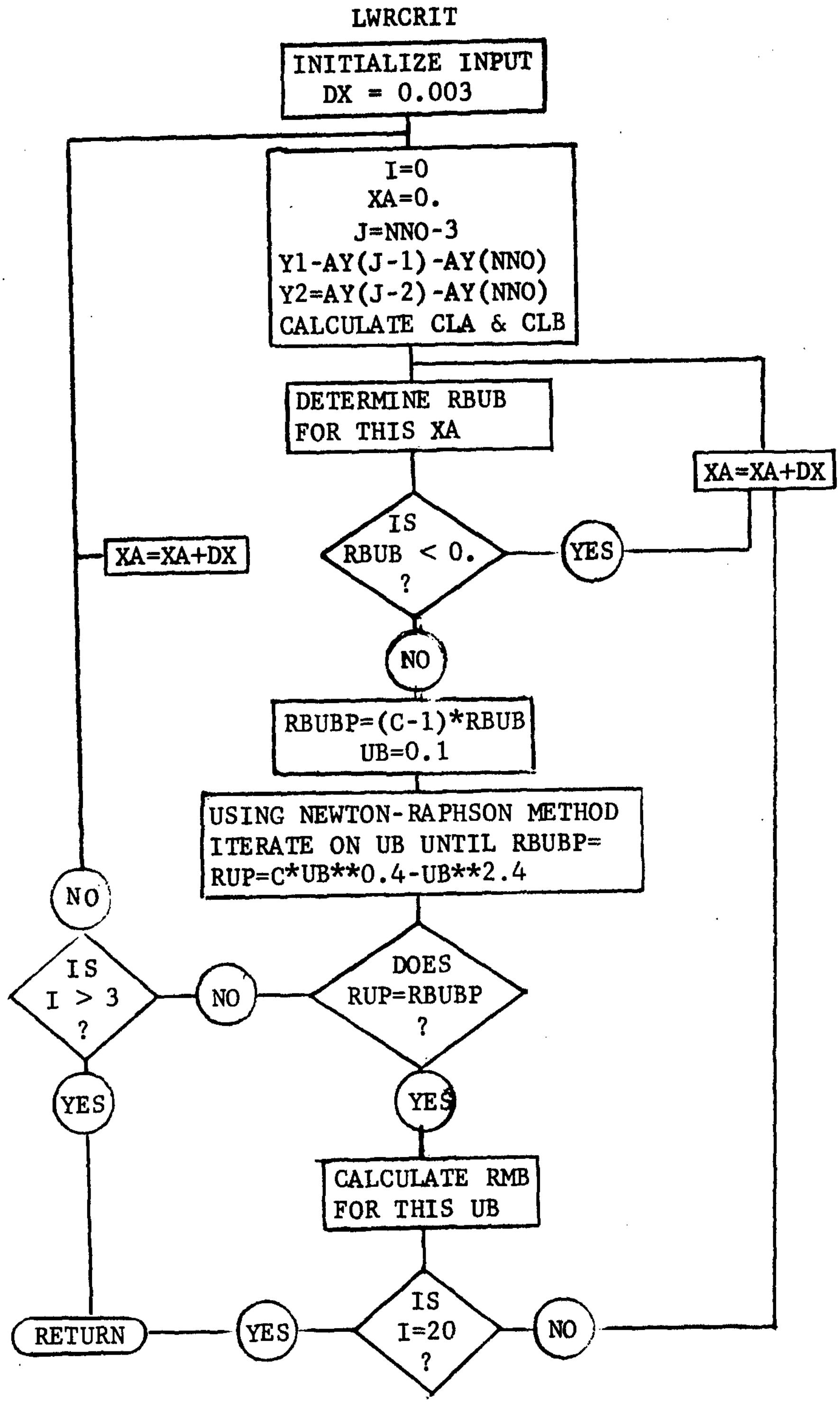

UPRINIT
INITIALIZE INPUT
DCSO = 0.02
DCS1 = 0.03
CALCULATE RBUB
IS RBUB < 0. ?
YES
NO
RBUBP = (C-1)RBUB
UB = 0.1
USING NEWTON-RAPHSON METHOD ITERATE ON UB UNTIL RBUBP = RUP = C*UB**.4-UB**2.4
DOES RUP = RBUBP ?
NO
YES
RETURN
CALCULATE RMB FOR UB
SPRCRTI

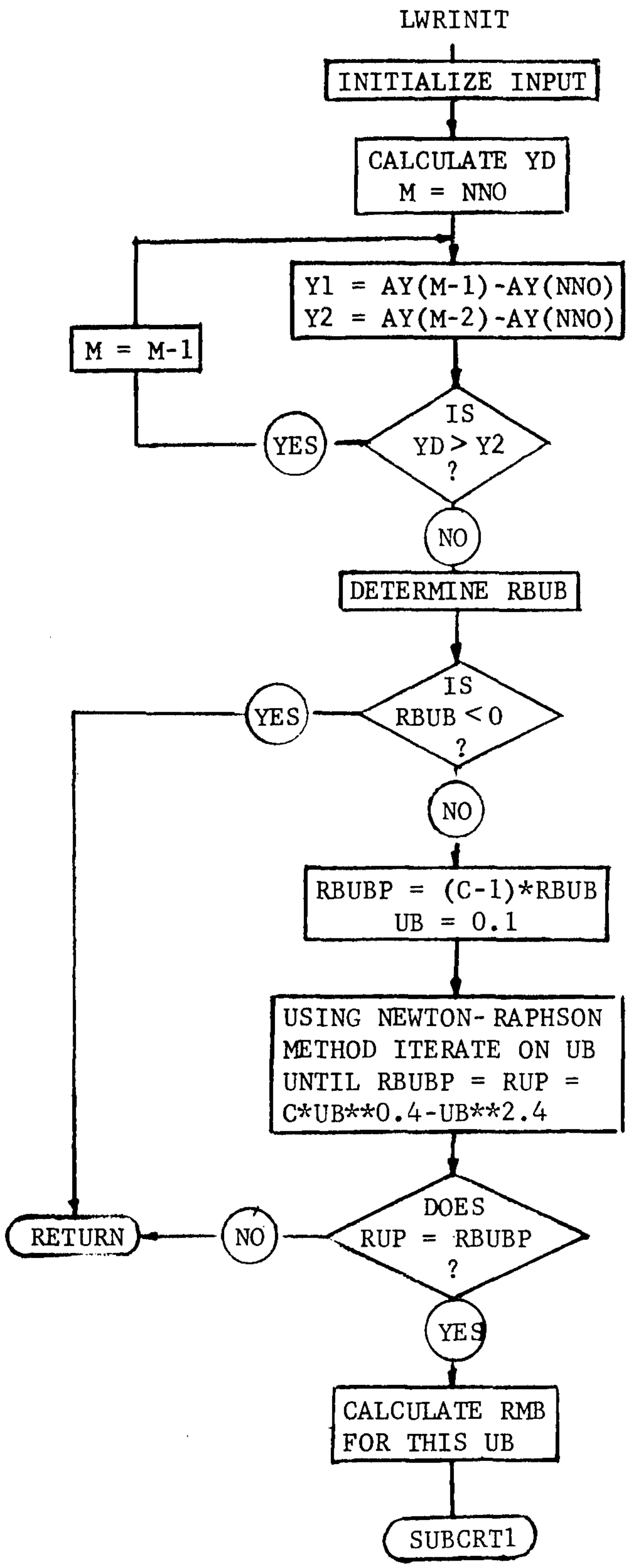
SPRCRT1

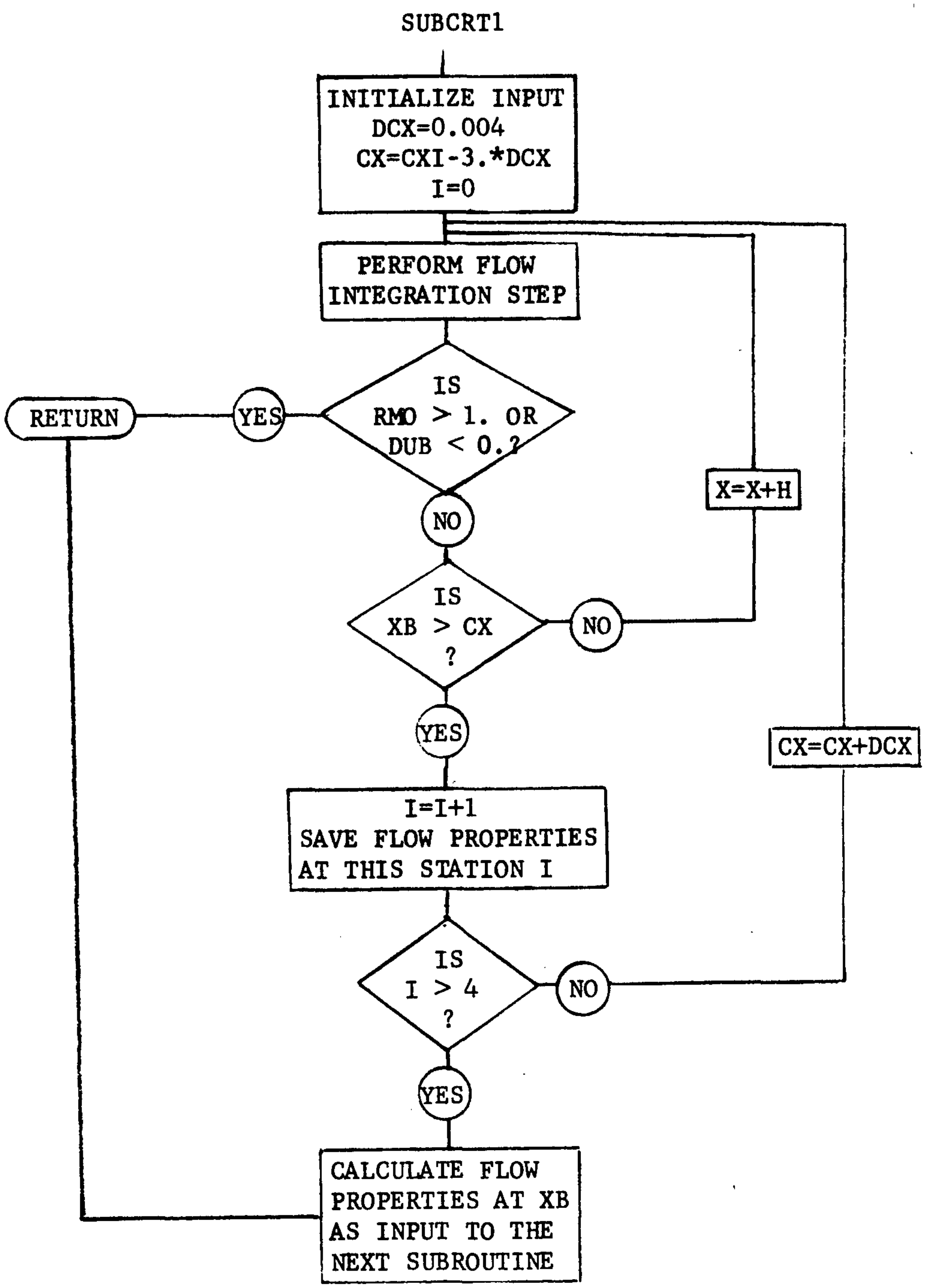

INITIALIZE INPUT
CSO = 0.9
DCSO = 0.02
DCS1 = 0.03
I = 0
PERFORM INTEGRATION STEP IN SUBSONIC REGION
X = X+H
IS RMB > CSO ?
NO
YES
CSO=CSO+DCSO
IS I = 0 ?
YES
HS=HS/2
H=H/2
NO
I = I + 1
SAVE FLOW PROPERTIES AT THIS STATION
IS I ≥ 4 ?
NO
YES
XA = XA+HX
CALCULATE RMB FOR NEW XA USING LGRNGN
IS RMB > 1.03 ?
NO
YES
COMPUTE FLOW PROPERTIES AT XA USING LGRNGN
b


b
I = 0
CS1 = RMB
PERFORM INTEGRATION STEP IN SUPERSONIC REGION
X=X+H
NO
IS RMB < 1. ?
CS1=CS1+DCS1
IS RMB > CS1 ?
NO
YES
YES
I = I + 1
SAVE FLOW PROPERTIES AT THIS STATION
NO
IS I ≥ 4 ?
YES
CALCULATE FLOW PROPERTIES AT X FOR INPUT TO THE NEXT SUBROUTINE
RETURN

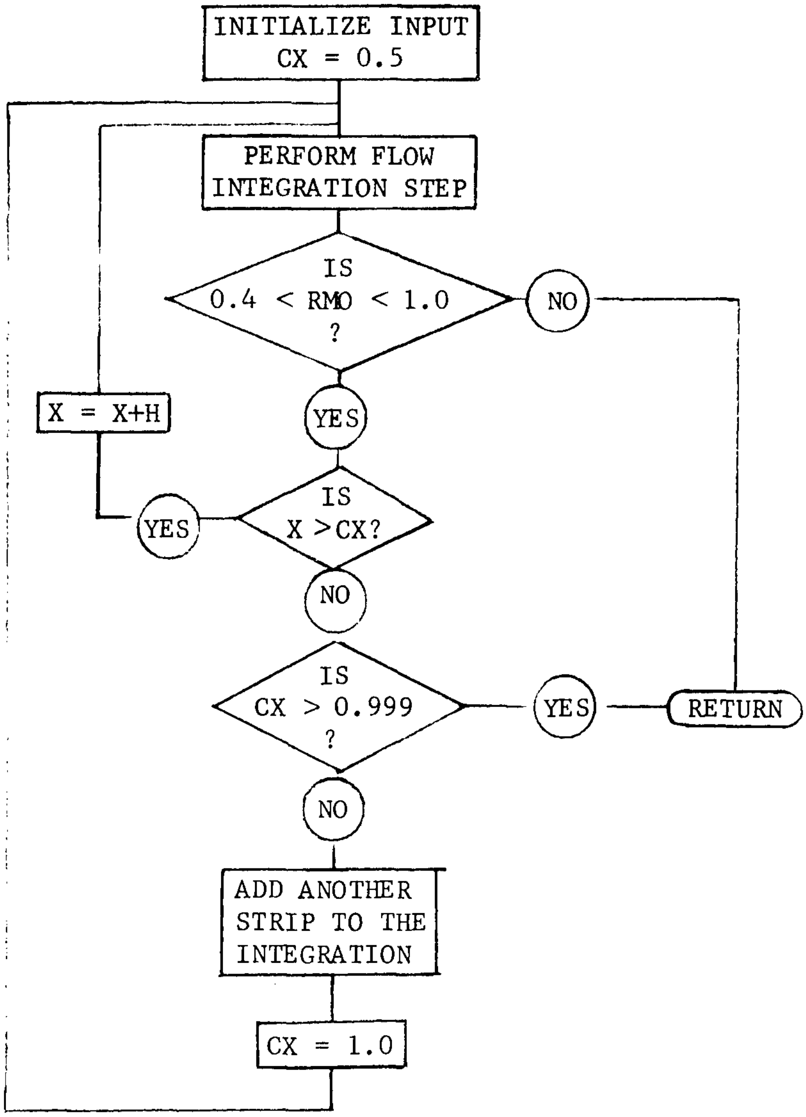

SPRCRT2
INITIALIZE INPUT
PERFORM FLOW INTEGRATION STEP IN SUPERSONIC REGION
IS DDQ > CDDQ ?
YES
NN=NC
NO
IS DUO > 100 OR RMO < 0.1?
YES
NO
IS X > 0.1 ?
YES
H = 0.005
NO
X=X+H
IS X > SL ?
NO
YES
APPLY RANKINE-HUGONIOT RELATIONS THROUGH SHOCK WAVE
PERFORM FLOW INTEGRATION STEP IN SUBSONIC REGION
IS DUO > 50 OR RMO > 1.?
YES
RETURN
NO
X=X+H
IS X > 1.0 ?
NO
YES
SAVE FLOW PARAMETERS CALCULATED AT FINAL STATION AS INPUT TO NEXT SUBROUTINE

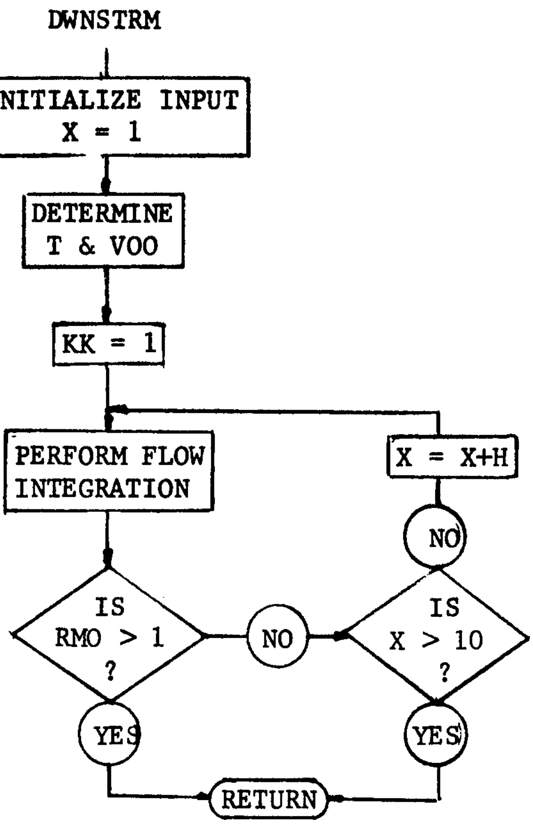

LWRCRIT
INITIALIZE INPUT
DX = 0.003
I=0
XA=0.
J=NNO-3
Y1-AY(J-1)-AY(NNO)
Y2=AY(J-2)-AY(NNO)
CALCULATE CLA & CLB
DETERMINE RBUB
FOR THIS XA
XA=XA+DX
IS
RBUB < 0.
?
YES
XA=XA+DX
NO
RBUBP=(C-1)*RBUB
UB=0.1
USING NEWTON-RAPHSON METHOD
ITERATE ON UB UNTIL RBUBP=
RUP=C*UB**0.4-UB**2.4
NO
IS
I > 3
?
NO
DOES
RUP=RBUBP
?
YES
YES
CALCULATE RMB
FOR THIS UB
RETURN
YES
IS
I=20
?
NO


LWRINIT
INITIALIZE INPUT
CALCULATE YD
M = NNO
Y1 = AY(M-1)-AY(NNO)
Y2 = AY(M-2)-AY(NNO)
M = M-1
IS
YD > Y2
?
YES
NO
DETERMINE RBUB
IS
RBUB < 0
?
YES
NO
RBUBP = (C-1)*RBUB
UB = 0.1
USING NEWTON-RAPHSON
METHOD ITERATE ON UB
UNTIL RBUBP = RUP =
C*UB**0.4-UB**2.4
DOES
RUP = RBUBP
?
NO
RETURN
YES
CALCULATE RMB
FOR THIS UB
SUBCRT1


SUBCRT1
INITIALIZE INPUT
DCX=0.004
CX=CXI-3.*DCX
I=0
PERFORM FLOW INTEGRATION STEP
IS RMO > 1. OR DUB < 0.?
YES
RETURN
NO
IS XB > CX ?
NO
X=X+H
YES
I=I+1
SAVE FLOW PROPERTIES AT THIS STATION I
IS I > 4 ?
NO
CX=CX+DCX
YES
CALCULATE FLOW PROPERTIES AT XB AS INPUT TO THE NEXT SUBROUTINE

SUBCRT2


INITIALIZE INPUT
CX = 0.5
PERFORM FLOW
INTEGRATION STEP
IS
0.4 < RMO < 1.0
?
NO
YES
X = X+H
IS
X >CX?
YES
NO
IS
CX > 0.999
?
YES
RETURN
NO
ADD ANOTHER
STRIP TO THE
INTEGRATION
CX = 1.0


DWNSTRM
INITIALIZE INPUT
X = 1
DETERMINE
T & VOO
KK = 1
PERFORM FLOW
INTEGRATION
X = X+H
NO
IS
RMO > 1
?
NO
IS
X > 10
?
YES
YES
RETURN

APPENDIX D

DESCRIPTION OF INTERACTIVE GRAPHICS PROGRAM

A description of the interactive graphics program requires knowledge of the Graphic Pac subroutines.* Such software is invaluable to the applications programmer because it permits him to use the interactive graphics facility with relative ease and simplicity.

Graphic Pac allows the applications programmer to create light buttons, light registers, and text entities which appear on the console screen. The working area of the console screen is a square inscribed within the circular screen area. The lower left-hand corner of the working area has the coordinates (-57, -57) and the upper right-hand corner has the coordinates (57, 57). The light buttons, light registers, and text entities can appear at any of about 13,000 addressable points on the screen. The light buttons and some of the text entities are light pen detectable (i.e., when the area of the screen where they appear is touched with the light pen, the graphics program executes the task overlay which is associated with them).

An inspection of the interactive graphics program listing could serve as an introduction to the workings of the program. The comment cards give a pretty good indication of the items currently being displayed.

Items are displayed by the statement

```
CALL GENDF(ID,0)
```

where ID is a six-integer array identifying a text or polyline entity. These polyline entities are created in previous statements. Three statements commonly used in conjunction are

```
CALL ENSHFT(12HY= ...,2,Y,7H(F10.6))
CALL MODFY(ID,1,2,12HY=       )
CALL GENDF(ID,0)
```

These statements take the value Y in format F10.6 and place it after the equals sign of Y= . The text entity ID is then modified to reflect this new information and then this text entity is displayed.

---

*See NSRDC Technical Note CMD 42-28, August 1973.

BLOCK DATA CGRAF

The block data program assigns six integer values to the arrays to identify each text or polyline entity. It also stores information used in some of the text entities and light registers to display currently computed values. Some of the identifying values in BLOCK DATA CGRAF may be changed during program execution, but most of the values are preserved.

SUBROUTINE PLOTT(X1MIN,X1MAX,Y1MIN,Y1MAX)

This subroutine creates two polyline entities which graphically display the data stored in arrays X1 and Y1. Two polyline entities are created to display these data because of the way in which they were obtained. For example, PROGRAM AFU2 calls this subroutine to display velocity gradients computed in the initial flow integration on the upper surface. Since velocity gradients cannot be computed at the sonic point, calculations are made to a point just before and a point just after the sonic point. The point at which one set of values ends and the other set begins is stored in the word NNI(7). Using this information, the two polyline entities are created and displayed in this subroutine.

The arguments of this subroutine are:

X1MIN = least upper bound for x scaling

X1MAX = greatest lower bound for x scaling

Y1MIN = least upper bound for y scaling

Y1MAX = greatest lower bound for y scaling

If only one value is stored in Y1, a large X covers the graphical display.

SUBROUTINE PLOTT1(X1MIN,X1MAX,Y1MIN,Y1MAX)

This subroutine creates one polyline entity which graphically displays the data stored in arrays X1 and Y1.

The arguments of this subroutine are:

X1MIN = least upper bound for x scaling

X1MAX = greatest lower bound for x scaling

Y1MIN = least upper bound for y scaling

Y1MAX = greatest lower bound for y scaling

If only one value is stored in Y1, a large X covers the graphical display.

SUBROUTINE PLOTT2(X1MIN,X1MAX,Y1MIN,Y1MAX,Y2MIN,Y2MAX,J)

This subroutine creates two polyline entities which graphically display the data stored in X1 and Y1 and X1 and Y2.

The arguments of this subroutine are:

X1MIN = least upper bound for X1 scaling

X1MAX = greatest lower bound for X1 scaling

Y1MIN = least upper bound for Y1 scaling

Y1MAX = greatest lower bound for Y1 scaling

Y2MIN = least upper bound for Y2 scaling

Y2MAX = greatest lower bound for Y2 scaling

J = an indication of whether or not Y2 values shall be displayed

If only one value is stored in array Y1, a large X covers the graphical display.

SUBROUTINE AMXMNI(Y1MAX,Y1MIN)

This subroutine determines the largest and smallest values stored in array Y1.

The arguments of this subroutine are:

Y1MAX = largest value stored in Y1

Y1MIN = smallest value stored in Y1

SUBROUTINE AMXMN2(Y2MAX,Y2MIN)

This subroutine determines the largest and smallest values stored in array Y2.

The arguments of this subroutine are:

Y2MAX = largest value stored in Y2

Y2MIN = smallest value stored in Y2

SUBROUTINE AREA1(XMIN,XMAX,YMIN,YMAX)

This subroutine defines a subscreen area which covers the area defined by the screen coordinates (-40, -40) and (57, 57) and creates a grid display for this subscreen area.

The arguments of this subroutine are:

XMIN = smallest x value

XMAX = largest x value

YMIN = smallest y value

YMAX = largest x value

SUBROUTINE AREA2(XMIN,XMAX,YMIN,YMAX,ID)

This subroutine defines a subscreen area and creates a grid display for this subscreen area.

The arguments of this subroutine are:

XMIN = smallest x value

XMAX = largest x value

YMIN = smallest y value

YMAX = largest y value

ID = subscreen area

One of two subscreen areas can be defined according to the values of ID. If ID = 1, the defined subscreen area is covered by screen coordinates (-40, -40) and (57, 10). If ID = 2, the defined subscreen area is covered by screen coordinates (-40, 17) and (57, 57).

PROGRAM LIEN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)

This program comprises the (0, 0) overlay and initiates execution of the graphic display files.

PROGRAM NPUT

This task reads input from punched cards and writes out some of the input on a line printer. The program writes out the label card which identifies the run and the data points and first derivatives at these data points which determine the airfoil shape being analyzed. Control is then automatically transferred to PROGRAM STUP.

PROGRAM STUP

This task creates the text entities and the light registers and light buttons used throughout the interactive graphics program.

The 59 data points required for the creation of the polyline entity NAIRFL are stored in X and Y arrays. When this entity is displayed, the airfoil shape is shown in the lower left-hand corner of the screen.

The text entities stored in COMMON blocks INPUT and NOUT are displayed in the lower right-hand corner of the screen. The 18 light registers corresponding to these text entities are displayed by using the same screen coordinates.

The text entities stored in COMMON blocks NPRCD are displayed in the lower left-hand corner of the screen. These entities display an asterisk which is either blinking or nonblinking. The nonblinking asterisks are light pen detectable and are capable of transferring program control to a particular task. The text entities stored in COMMON block NAXES are displayed in conjunction with the displays of the graphed output of a particular task.

PROGRAM STRT

This task displays the flow conditions in the lower right-hand corner of the screen. The flow conditions are displayed in the four text entities and are light pen detectable. The four text entities contain information in the free-stream Mach number, the angle of attack, and location of the outermost strip on the upper and lower surfaces. Two light buttons are also enabled, one of which allows program control to transfer to PROGRAM PRP1. The program is terminated by a call to subroutine WAITE and awaits an attention interrupt from an enabled attention source.

PROGRAM AFU1

This task erases all previous displays and displays information from subroutine IOUPRCT. The displayed text entities allow program control to be transferred to PROGRAM PRP3 or PRP4, or to PROGRAM STRT or PRP2. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM PRP3

This task retrieves the integer array of the text entity from which program control was transferred. If ID(3) = 1, LRSUPR was the attention source and if ID(3) = 2, LRSUB was the attention source. If ID(3) = 2, the program awaits an attention interrupt. If ID(3) = 1, the program erases all previous displays and displays the text entities and light buttons associated with PROGRAM AFU2. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM AFU2

This task displays information from subroutine IOUPRIN corresponding to initial flow integration on the upper surface. The two text entities display values of RBUB and UB, and the polyline entity graphically displays DUDX versus x. If the flow integration is complete, NN2 = 1, and the text entity which allows program control to be transferred to task PRP5 is displayed. The other two displayed text entities allow program control to be transferred to task STRT or PRP2. Two light buttons are also enabled, one of which allows program control to be transferred to task AFU2. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM AFL1

This task erases all previous displays and displays information from subroutine IOLWRCT. The displayed text entities allow program control to be transferred to task PRP3 or PRP4, or to task STRT or PRP2. The

program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM PRP4

This task retrieves the integer array of the text entity from which program control was transferred. If ID(3) = 1, LRSUPR was the attention source, and if ID(3) = 2, LRSUB was the attention source. If ID(3) = 1, the program awaits an attention interrupt. The task erases all previous displays and displays the text entities and light buttons associated with task AFL2. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM AFL2

This task displays information from subroutine IOLWRIN corresponding to flow integration on the lower surface. The three text entities display values of RBUB and UB and PO at the trailing edge, and the polyline entity graphically displays M_o versus x. If the flow integration is complete, NN2 = 1, and the text entity which allows program control to be transferred to task AFU1 is displayed. If NN2 = 1, the word IGO(J) = 1, and if IGO(J) = 1 for both upper and lower surfaces, the text entity which allows program control to be transferred to task PRP6 is displayed. The other two displayed text entities allow program control to be transferred to task STRT or PRP2. Two light buttons are also enabled, one of which allows program control to be transferred to task AFL2. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM PRP5

This task erases all previous displays and displays the text entities and light buttons associated with task AFU3. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM AFU3

This task displays information from subroutine IOSPCT2 corresponding to flow integration on the upper surface. The text entity displays the value of PO at the trailing edge and the polyline entities graphically display MO versus X. If the flow integration is complete, NN2 = 1, and the text entity which allows program control to be transferred to task AFL1 is displayed. If NN2 = 1, the word IGO(J) = 1, and if IGO(J) = 1 for both upper and lower surface, the text entity which allows program control to be transferred to task PRP6 is displayed. The three other text entities allow program control to be transferred to tasks PRP2, AFL1, AFU1, or PRP3. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM PRP6

This task erases all previous displays and displays the text entities and light buttons associated with task DWN1. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM DWN1

This task displays information from subroutine IODNSTM corresponding to downstream flow integration. The polyline entities display PO and P1 versus X. If flow integration is complete NN2 = 1, and the text entity which allows program control to be transferred to task DWN2 is displayed. The other four displayed text entities allow program control to be transferred to task PRP2, AFL1, or PRP5. Two light buttons are also enabled which allow program control to be transferred to task DWN1. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM DWN2

This task erases all previous displays and displays information from subroutine AKUTTA which contains the pressure distributions on the upper and lower surfaces. Three text entities are displayed which allow program

control to be transferred to tasks PRP2, AFL1, and AFU1. The program terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM CVLI

This task retrieves the integer array of the text entity from which program control was transferred. The text entity is erased and replaced by a light register which has the same code number as the first integer of the text entity. The task then awaits keyboard information to be typed in. When a new value is typed in and the keyboard release button is activated, the light register will be replaced with a text entity with the typed-in value. The task terminates and awaits an attention interrupt from an enabled attention source.

PROGRAM CVLR

This task is structured identically to task CVLI. The keyboard information is typed in under a different format, however.

PROGRAM STOP

This task erases the screen display and releases the console from the computer.

PROGRAM CHGV

This task computes a new value for the normal velocity component of the innermost strip according to LL(J). If J = 2, the Lagrangian function is used, and if J = 3, the parabolic function is used.

APPENDIX E

INPUT DESCRIPTION FOR INTERACTIVE GRAPHICS

CARD 1 FORMAT(I2,8X,7A10)

Columns	Description
1-2	Console number
11-80	Title for the run

CARD 2 FORMAT(2I2)

Columns	Description	
1-2	NP(1)	number of data points on upper surface of airfoil
3-4	NP(2)	number of data points on lower surface of airfoil

CARDS SET 1 FORMAT(3F20.15)

Columns	Description	
1-20	XX(1,1)	x-coordinate
21-40	YY(1,1)	y-coordinate
41-60	AM(1,1)	first derivative at this point

CARDS SET 2

Columns	Description	
1-20	XX(1,2)	x-coordinate on lower surface
21-40	YY(1,2)	y-coordinate on lower surface
41-60	AM(1,2)	first derivative at this point

Format is repeated for remainder of airfoil data points

CARD SET 3

CARD 1

Columns	Description	
1-10	blank	
11-20	DVOOI	initial cross velocity gradient
21-30	XS	stagnation point
31-40	XAO	initial point upper surface

CARD SET 3 - CARD 1 (cont.)

Columns	Description	
41-50	CYDU	flow parameter upper surface
51-60	XAI	initial point lower surface
61-70	CYDL	flow parameter lower surface
71-80	SL	shock location

CARD 2

Columns	Description	
1-10	XOO	distance from first station of upstream integration to airfoil surface
11-20	RMT	Mach number used in upstream flow integration, $\approx 0.95\ M_\infty$
21-30	CDY $\approx$ 0.1	
31-40	YU	location of outermost strip on upper surface
41-50	YL	location of outermost strip on lower surface
51-60	CS	a point in subcritical flow calculations where a new integration scheme is adopted
61-70	RMC $\approx$ 0.92	Mach number upper limit for subroutine SPRCRT1
71-80	BETAD	angle of shock foot for subroutine SPRCRT2

CARD 3 FORMAT(8F10.6)

Columns	Description	
1-10	DELS	0 for isentropic flow and 1 for nonisentropic flow through shock foot in subroutine SPRCRT2
11-20	CDDQ $\approx$ 11.0	upper limit for DDQ in subroutine SPRCRT2
21-30	RKI $\approx$ 5.0	exponent for normal velocity component in downstream integration

CARD 4 FORMAT(6I1,4X,7F10.6)

Columns	Description	
1	NN1	number of strips used in upstream integration
2	NA2	number of additional strips used near final station of upstream integration
3	NN3	number of strips used in initial solution of upper surface flow integration
4	NN4	number of strips used in initial solution of lower surface flow integration
5	NN5	number of strips used in flow solution along the upper surface
6	NN6	intermediate strip used in calculating downstream flow conditions

EXAMPLE INPUT/OUTPUT

This example corresponds to the output which was given for the demonstration of the interactive graphics program for an advanced airfoil. In a real situation, however, there would be a greater amount of output corresponding to the trial solutions for the various solution processes. The input and output for this example represent only the solutions which are known to be correct.

The following represents a deck for a graphics run:

```
CXXX,CM20000,P4,G.
CHARGE,CXXX,000000000.
ATTACH(IGSGO,CAMVIGSGO)
ATTACH(ATSK,CXXXATSK)
IGSGO(ATSK,1)
7/8/9
01        TRANSONIC FLOW PAST AN ADVANCED AIRFOIL
0.7       1.5
          6.0      .004     .0125    1.564133       .018     .6946    .485
2.5       .68      .1       7.0      7.0            .045     .92      90.0
0.0       11.0     5.0
.536653   0.02     .0002    .002     .005                    .02
2222
0.0                0.0               5.40000000000000
.0125              .0304             .879686323493225
```

.05	.0519	.370509412053975
.1	.06529	.194774312423792
.1499999999999999	.07893	.098052334572903
.249999999999999	.0832	.073397323457582
.299999999999999	.0863	.050558371596735
.349999999999998	.08826	.0279691901555
.399999999999999	.08912	.006764867781260
.449999999999998	.08912	.006764867781280
.449999999999998	.08896	.013028661280615
.499999999999998	.08782	.032650222658823
.549999999999997	.08568	.053170448084134
.599999999999998	.08247	.075067985004662
.649999999999996	.0723	.132701567406445
.749999999999996	.06476	.169436118476982
.799999999999997	.05533	.207753958685581
.849999999999998	.04899	.245748046780653
.899999999999995	.08078	.282253854191833
.949999999999996	.0159	.310636536452030
.999999999999996	0.0	.321999999999999
0.0	0.0	5.050000000000011
.0125	.03	.992163451845627
.050	.05333	.379092385234589
.1	.06639	.180750933111021
.149999999999999	.07356	.111703882321330
.199999999999999	.07802	.069633537603695
.249999999999999	.0807	.038161967263869
.299999999999999	.08193	.012318593340822
.349999999999998	.08199	.009436340627176
.399999999999999	.08096	.032773230832096
.449999999999998	.07865	.060470736044399
.499999999999998	.0748	.094943824990326
.549999999999997	.069	.13875396399439
.599999999999998	.0607	.196040319032322
.649999999999995	.0495	.247084759876374
.699999999999996	.0366	.261620641462178

.749999999999996	.024	.236432674274866
.799999999999997	.0134	.184648661438304
.849999999999998	.00581	.1163726799718
.899999999999995	.002	.033860618674168
.949999999999996	.0026	.059215154668539
.999999999999996	.00805	.16

****

TRANSONIC FLOW PAST AN ADVANCED AIROFOIL

X(UPPER)	Y(UPPER)	DY/DX(UPPER)	X(LOWER)	Y(LOWER)	DY/DX(LOWER)
0.000000000000	0.000000000000	5.400000000000	0.000000000000	0.000000000000	5.050000000000
.012500000000	.030400000000	.879686323493	.012500000000	.030000000000	.992163451846
.050000000000	.051900000000	.370509412054	.050000000000	.053330000000	.379092385235
.100000000000	.065290000000	.194774312424	.100000000000	.066400000000	.180750933111
.150000000000	.073250000000	.131393338251	.150000000000	.073560000000	.111703882321
.200000000000	.078930000000	.098052334573	.200000000000	.078020000000	.069633537604
.250000000000	.083200000000	.073397323458	.250000000000	.080700000000	.038161967264
.300000000000	.086300000000	.050558371597	.300000000000	.081930000000	.012318593341
.350000000000	.088260000000	.027969190156	.350000000000	.082000000000	-.009436340627
.400000000000	.089120000000	.006764867781	.400000000000	.080960000000	-.032773230832
.450000000000	.088960000000	-.013028661281	.450000000000	.078650000000	-.060470736044
.500000000000	.087820000000	-.032650222659	.500000000000	.074800000000	-.094943824990
.550000000000	.085680000000	-.053170448084	.550000000000	.069000000000	-.138753963994
.600000000000	.082480000000	-.075067985005	.600000000000	.060700000000	-.196040319032
.650000000000	.078110000000	-.100757611897	.650000000000	.049500000000	-.247084759876
.700000000000	.072300000000	-.132701567406	.700000000000	.036600000000	-.261620641462
.750000000000	.064760000000	-.169436118477	.750000000000	.024000000000	-.236432674275
.800000000000	.055330000000	-.207753958686	.800000000000	.013400000000	-.184648661438
.850000000000	.043990000000	-.245748046781	.850000000000	.005810000000	-.116372679972
.900000000000	.030780000000	-.282253854192	.900000000000	.002000000000	-.033860618674
.950000000000	.015900000000	-.310636536452	.950000000000	.002600000000	.059215154669
1.000000000000	0.000000000000	-.322000000000	1.000000000000	.008050000000	.160000000000

MACH NO.= .700000 ALPHA= 1.500000

** UPSTREAM-SOLUTION ***

NN = 5, NA = 3

DVOO(I) = 6.0000, XOO = 2.5000, RMT = .6800, CDY = .1000, H = .0200

YINF(UPPER) = 7.0000, YINF(LOWER) = 7.0000

X	MO	X	MO
0.000000	.700000	1.580000	.664379
.060000	.700000	1.640000	.662260
.120000	.699999	1.700000	.660010
.180000	.699991	1.760000	.657632
.240000	.699963	1.820000	.655128
.300000	.699888	1.880000	.652501
.360000	.699720	1.940000	.649753
.420000	.699396	1.955000	.647770
.480000	.698823	1.970000	.644476
.540000	.697881	1.985000	.639939
.600000	.696420	2.000000	.634252
.660000	.694254	2.015000	.627485
.720000	.691164	2.030000	.619631
.780000	.686901	2.045000	.610563
.840000	.681189	2.060000	.599998
.860000	.678913	2.075000	.587490
.920000	.678489	2.090000	.572452
.980000	.677921	2.105000	.554204
1.040000	.677209	2.120000	.532026
1.100000	.676353	2.135000	.505221
1.160000	.675353	2.150000	.473153
1.220000	.674210	2.160000	.448560
1.280000	.672923	2.175000	.406421
1.340000	.671493	2.180000	.390875
1.400000	.669923	2.195000	.339260
1.460000	.668212	2.201000	.316505
1.520000	.666363	2.207000	.292347

Y	M	Y	M
7.000000	.700000	.234647	.730981
3.502907	.701421	.128829	.733025
1.754687	.700521	.076939	.727602
.883280	.704802	.001139	.292347
.451885	.714635		

DE = .056000 YSO = .012598

INITIAL VELOCITY PROFILE

UPPER SURFACE			LOWER SURFACE		
Y	U	V	Y	U	V
7.000000	1.000000	0.000000	7.000000	1.000000	0.000000
3.502907	1.001827	.006582	3.502907	1.001580	.006585
1.754687	1.000620	.010738	1.754687	1.000252	.010738
.883280	1.006074	.018385	.883282	1.005286	.018396
.451885	1.018548	.029913	.451889	1.017053	.029932
.234647	1.037219	.076927	.234635	1.034252	.077056
.128829	1.031526	.152198	.128809	1.027078	.152380
.076939	1.010017	.229310	.076917	1.004665	.229294
.001139	.432725	.032349	-.001139	.432725	-.032349

STAGNATION SOLUTION

XS = .004000

FROM THE UPSTREAM SOLUTION, DE = .056000, YSO = .012598

AIRFOIL COORDINATES

X	Y(UPPER)	Y(LOWER)	X	Y(UPPER)	Y(LOWER)
0.000000	0.000000	0.000000	.015000	.032128	-.032781
.000600	.003099	-.002934	.015600	.032605	-.033348
.001200	.005957	-.005655	.016200	.033073	-.033907
.001800	.008587	-.008172	.016800	.033535	-.034457
.002400	.010999	-.010496	.017400	.033989	-.034998
.003000	.013205	-.012637	.018000	.034437	-.035530
.003600	.015218	-.014605	.018600	.034877	-.036054
.004200	.017049	-.016411	.019200	.035311	-.036570
.004800	.018710	-.018065	.019800	.035737	-.037077
.005400	.020212	-.019577	.020400	.036157	-.037577
.006000	.021568	-.020958	.021000	.036570	-.038068
.006600	.022788	-.022217	.021600	.036977	-.038552
.007200	.023885	-.023366	.022200	.037378	-.039027
.007800	.024871	-.024415	.022800	.037772	-.039495
.008400	.025756	-.025373	.023400	.038159	-.039956
.009000	.026554	-.026251	.024000	.038541	-.040409
.009600	.027275	-.027059	.024600	.038917	-.040855
.010200	.027932	-.027809	.025200	.039287	-.041293
.010800	.028536	-.028509	.025800	.039650	-.041725
.011400	.029099	-.029171	.026400	.040009	-.042150
.012000	.029632	-.029804	.027000	.040361	-.042567
.012600	.030148	-.030419	.027600	.040708	-.042979
.013200	.030654	-.031023	.028200	.041049	-.043383
.013800	.031153	-.031618	.028800	.041385	-.043781
.014400	.031645	-.032204	.029400	.041716	-.044173

STAGNATION STREAMLINE

X	Y	X	Y
-.052000	-.027285	-.018400	-.022703
-.046400	-.026875	-.012800	-.021328
-.040800	-.026346	-.007200	-.019735
-.035200	-.025681	-.001600	-.017906
-.029600	-.024863	.004000	-.015826
-.024000	-.023876		

YS = .015826 DVOO(F) = 6.018845

TEST OF CRITICALITY

UPPER SURFACE

FROM THE UPSTREAM SOLUTION, NN = 9, DE = .056000

FROM THE STAGNATION SOLUTION, YS = .015826, YSO = .012598

XB	MB	XB	MB
.0006	.3663	.0033	.6187
.0011	.4120	.0039	.6844
.0017	.4592	.0044	.7661
.0022	.5086	.0050	.9021
.0028	.5612		

** INITIAL SOLUTION **

UPPER SURFACE

NN = 6, XAO = .012500, CYO = 1.56413300, RMC = .920000, HS = .000200

RBUB = .984972 UB = .971579

XB	MB	PB	DUDX
.0138	.6857	1.0127	16.2850
.0142	.6932	1.0061	16.3241
.0147	.7007	.9994	16.3604
.0152	.7083	.9926	16.3938
.0157	.7159	.9858	16.4243
.0162	.7235	.9789	16.4518
.0166	.7312	.9720	16.4763
.0171	.7389	.9651	16.4977
.0176	.7467	.9581	16.5159
.0181	.7544	.9511	16.5310
.0186	.7622	.9441	16.5429
.0191	.7701	.9370	16.5516
.0196	.7779	.9299	16.5570
.0201	.7858	.9228	16.5591
.0206	.7937	.9156	16.5578
.0211	.8017	.9084	16.5532
.0216	.8097	.9012	16.5452
.0221	.8176	.8940	16.5337
.0226	.8257	.8868	16.5187
.0231	.8337	.8796	16.5001
.0236	.8417	.8723	16.4780
.0241	.8498	.8651	16.4522
.0247	.8579	.8578	16.4227
.0252	.8660	.8505	16.3894
.0257	.8741	.8433	16.3521
.0262	.8822	.8360	16.3106
.0267	.8903	.8288	16.2649
.0273	.8984	.8215	16.2144
.0274	.9011	.8191	16.1965
.0277	.9052	.8155	16.1947
.0280	.9093	.8119	16.1691
.0282	.9134	.8083	16.1427
.0285	.9174	.8047	16.1153
.0287	.9202	.8023	16.0965
.0289	.9242	.7987	16.0676
.0292	.9283	.7951	16.0378
.0295	.9324	.7915	16.0070
.0297	.9365	.7879	15.9752
.0300	.9406	.7843	15.9425
.0303	.9446	.7807	15.9087

XB	MB	PB	DUDX
.0305	.9487	.7772	15.8738
.0308	.9528	.7736	15.8377
.0311	.9569	.7700	15.8002
.0313	.9609	.7665	15.7613
**********	**********	**********	**********
.0362	1.0328	.7049	15.0170
.0367	1.0409	.6982	14.9235
.0373	1.0489	.6915	14.8216
.0378	1.0569	.6849	14.7189
.0382	1.0622	.6805	14.6504
.0388	1.0701	.6740	14.5481
.0393	1.0780	.6675	14.4477
.0399	1.0859	.6610	14.3513
.0403	1.0912	.6568	14.2907
.0408	1.0991	.6504	14.2089
.0412	1.1043	.6462	14.1639
.0418	1.1020	.6480	-3.9720
.0423	1.1000	.6496	-3.3607
.0429	1.0984	.6509	-2.7361
.0434	1.0971	.6520	-2.1054
.0440	1.0962	.6527	-1.4765
.0446	1.0956	.6532	-.8575
.0451	1.0953	.6534	-.2558
.0457	1.0954	.6534	.3178
.0463	1.0958	.6531	.8593
.0468	1.0965	.6525	1.3616
.0474	1.0974	.6518	1.8201
.0480	1.0985	.6508	2.2314
.0485	1.0999	.6497	2.5934
.0491	1.1015	.6485	2.9053
.0497	1.1032	.6471	3.1670
.0503	1.1050	.6456	3.3795
.0508	1.1070	.6441	3.5442
.0514	1.1090	.6424	3.6628
.0520	1.1111	.6408	3.7624
.0526	1.1132	.6391	3.8555
.0531	1.1154	.6373	3.9400
.0537	1.1176	.6355	4.0163
.0543	1.1199	.6337	4.0847
.0545	1.1207	.6331	4.1058

AIRFOIL SOLUTION
UPPER SURFACE

NC = 5

SHOCK LOC. = .485000, BETA = 90.000000, DELS = 0.000000, CDDQ = 11.000000, HO = .003000

FROM INITIAL CONDITIONS, NN = 6, X(INIT) = .053100

INTERMEDIATE VELOCITY DISTRIBUTION USING LAGRANGIAN FUNCTION

Y	U	V	Y	U	V
6.971576	1.000000	0.000000	.856865	1.026044	.017403
3.475218	1.006400	.006529	.426659	1.083837	.183356
1.727478	1.007842	.010959	.051615	1.426226	.463849

X	MO	PO	DUDX	DDQ	X	MO	PO	DUDX	DDQ
.0621	1.1373	.6200	1.1337	-3.7580	.4103	1.3150	.4905	-.0842	-.3130
.0711	1.1585	.6035	1.1038	-3.1607	.4253	1.3022	.4991	-.0895	-.3103
.0800	1.1817	.5858	1.0809	-2.3741	.4403	1.2890	.5081	-.0946	-.3077
.0890	1.2057	.5678	1.0647	-1.4564	.4553	1.2756	.5175	-.0997	-.3051
.0980	1.2298	.5501	1.0570	-.4454	.4706	1.2617	.5272	-.1048	-.3058
.1108	1.2634	.5260	1.0611	.5477	.4856	1.2475	.5373	-.1097	-.3073
.1258	1.3004	.5003	1.0742	1.3344	.5156	.8136	.8977	-.0439	0.0000
.1408	1.3353	.4770	1.0996	2.2770	.5456	.8112	.8998	-.0527	0.0000
.1558	1.3677	.4561	1.1405	3.4859	.5756	.8069	.9037	-.0617	0.0000
.1706	1.3972	.4376	1.2001	4.8571	.6056	.8008	.9093	-.0706	0.0000
.1856	1.4237	.4215	1.2865	7.0045	.6361	.7927	.9165	-.0802	0.0000
.2006	1.4470	.4078	1.4200	10.9717	.6661	.7827	.9256	-.0905	0.0000
.2156	1.4670	.3963	1.6563	20.0125	.6961	.7708	.9363	-.1016	0.0000
.2306	1.4558	.4027	-.0144	-.3461	.7261	.7570	.9488	-.1133	0.0000
.2456	1.4446	.4092	-.0206	-.3391	.7566	.7414	.9629	-.1247	0.0000
.2606	1.4334	.4158	-.0266	-.3333	.7866	.7244	.9782	-.1355	0.0000
.2756	1.4220	.4225	-.0326	-.3346	.8166	.7062	.9945	-.1448	0.0000
.2906	1.4106	.4294	-.0385	-.3358	.8471	.6871	1.0115	-.1525	0.0000
.3056	1.3991	.4364	-.0445	-.3369	.8771	.6674	1.0290	-.1585	0.0000
.3203	1.3875	.4436	-.0503	-.3269	.9071	.6471	1.0468	-.1622	0.0000
.3353	1.3758	.4509	-.0562	-.3342	.9376	.6266	1.0646	-.1632	0.0000
.3503	1.3640	.4584	-.0620	-.3325	.9676	.6060	1.0823	-.1603	0.0000
.3653	1.3520	.4661	-.0677	-.3289	.9981	.5857	1.0996	-.1535	0.0000
.3803	1.3399	.4740	-.0733	-.3237	1.0106	.5790	1.1052	-.1505	0.0000
.3953	1.3276	.4821	-.0788	-.3182					

DOWNSTREAM SOLUTION

NN = 3, H = .020000, RK = 5.000000

FROM UPPER SURFACE INTEGRATION, SHOCK LOC. = .4850, BETA = 90.0000, CS = 1.0000, CZ = 11.2041

X	MO	PO	P1
1.0800	.5775	1.1065	.9936
1.1600	.5759	1.1078	.9932
1.2400	.5742	1.1093	.9927
1.3200	.5724	1.1108	.9922
1.4000	.5704	1.1125	.9918
1.4800	.5683	1.1142	.9913
1.5600	.5662	1.1160	.9908
1.6400	.5639	1.1179	.9902
1.7200	.5615	1.1198	.9897
1.8000	.5591	1.1219	.9892
1.8800	.5565	1.1240	.9886
1.9600	.5538	1.1262	.9880
2.0400	.5510	1.1285	.9874
2.1200	.5481	1.1308	.9868
2.2000	.5452	1.1333	.9862
2.2800	.5421	1.1358	.9855
2.3600	.5389	1.1384	.9848
2.4400	.5356	1.1411	.9841
2.5200	.5322	1.1438	.9833
2.6000	.5287	1.1466	.9826
2.6800	.5251	1.1495	.9817
2.7600	.5214	1.1525	.9808
2.8400	.5176	1.1555	.9799
2.9200	.5136	1.1586	.9790
3.0000	.5096	1.1618	.9779
3.0800	.5055	1.1651	.9768
3.1600	.5012	1.1684	.9757
3.2400	.4968	1.1718	.9745
3.3200	.4924	1.1753	.9732
3.4000	.4878	1.1788	.9718
3.4800	.4831	1.1824	.9703
3.5600	.4783	1.1861	.9688
3.6400	.4733	1.1898	.9671
3.7200	.4683	1.1936	.9654
3.8000	.4632	1.1974	.9635
3.8800	.4580	1.2013	.9616
3.9600	.4527	1.2052	.9595
4.0400	.4473	1.2091	.9573
4.1200	.4418	1.2131	.9550
4.2000	.4363	1.2171	.9525
4.2800	.4307	1.2211	.9500
4.3600	.4250	1.2251	.9472
4.4400	.4192	1.2291	.9444
4.5200	.4135	1.2331	.9414
4.6000	.4076	1.2371	.9382
4.6800	.4018	1.2411	.9349
4.7600	.3959	1.2451	.9315
4.8400	.3900	1.2490	.9279
4.9200	.3841	1.2529	.9241
5.0000	.3782	1.2567	.9202
5.0800	.3723	1.2605	.9162
5.1600	.3665	1.2642	.9120
5.2400	.3607	1.2679	.9076
5.3200	.3549	1.2714	.9032
5.4000	.3493	1.2749	.8986
5.4800	.3437	1.2783	.8938
5.5600	.3382	1.2815	.8890

X	MO	PO	P1
5.6400	.3328	1.2847	.8840
5.7200	.3275	1.2878	.8790
5.8000	.3224	1.2907	.8739
5.8800	.3175	1.2935	.8688
5.9600	.3127	1.2962	.8636
6.0400	.3080	1.2988	.8584
6.1200	.3036	1.3012	.8532
6.2000	.2993	1.3035	.8481
6.2800	.2953	1.3056	.8430
6.3600	.2915	1.3077	.8381
6.4400	.2879	1.3095	.8334
6.5200	.2845	1.3113	.8289
6.6000	.2814	1.3129	.8247
6.6800	.2785	1.3143	.8208
6.7600	.2759	1.3157	.8172
6.8400	.2735	1.3169	.8141
6.9200	.2713	1.3179	.8114
7.0000	.2694	1.3189	.8092
7.0800	.2676	1.3197	.8074
7.1600	.2661	1.3205	.8062
7.2400	.2648	1.3211	.8053
7.3200	.2636	1.3217	.8049
7.4000	.2626	1.3222	.8047
7.4800	.2618	1.3226	.8049
7.5600	.2611	1.3229	.8052
7.6400	.2604	1.3232	.8056
7.7200	.2599	1.3234	.8061
7.8000	.2595	1.3237	.8065
7.8800	.2591	1.3238	.8069
7.9600	.2588	1.3240	.8071
8.0400	.2585	1.3241	.8072
8.1200	.2582	1.3242	.8071
8.2000	.2580	1.3244	.8069
8.2800	.2578	1.3244	.8064
8.3600	.2577	1.3245	.8057
8.4400	.2575	1.3246	.8049
8.5200	.2575	1.3246	.8038
8.6000	.2575	1.3246	.8027
8.6800	.2575	1.3246	.8015
8.7600	.2577	1.3245	.8003
8.8400	.2579	1.3244	.7991
8.9200	.2583	1.3242	.7981
9.0000	.2588	1.3240	.7973
9.0800	.2594	1.3237	.7969
9.1600	.2602	1.3233	.7969
9.2400	.2613	1.3228	.7974
9.3200	.2625	1.3222	.7984
9.4000	.2640	1.3215	.8000
9.4800	.2656	1.3207	.8022
9.5600	.2676	1.3198	.8050
9.6400	.2697	1.3187	.8083
9.7200	.2721	1.3175	.8120
9.8000	.2748	1.3162	.8162
9.8800	.2777	1.3148	.8206
9.9600	.2808	1.3132	.8254
10.0200	.2841	1.3115	.8303

TEST OF CRITICALITY
LOWER SURFACE

FROM THE UPSTREAM SOLUTION, NN = 9, DE = .056000

FROM THE STAGNATION SOLUTION, YS = .015826, YSO = .012598

XB	MB	XB	MB
.0055	.1882	.0171	.6167
.0083	.3657	.0200	.6923
.0112	.4681	.0230	.7805
.0141	.5434	.0259	.9460

AIRFOIL SOLUTION
LOWER SURFACE

NB = 6

XA = .0180, CYD = .6946, CX = .0450, HSO = .0020, HO = .0050

RBUB = .790322 UB = .699485

XB	MB	PB	DUDX	XB	MB	PB	DUDX
.0185	.4892	1.1777	7.9648	.0328	.5799	1.1045	5.6320
.0200	.5005	1.1690	7.8586	.0345	.5875	1.0981	5.1569
.0215	.5116	1.1602	7.7195	.0362	.5944	1.0923	4.6380
.0231	.5225	1.1516	7.5449	.0380	.6004	1.0871	4.0781
.0246	.5331	1.1430	7.3325	.0397	.6056	1.0827	3.4817
.0262	.5434	1.1347	7.0798	.0415	.6099	1.0791	2.8547
.0278	.5533	1.1266	6.7849	.0433	.6131	1.0762	2.2047
.0294	.5628	1.1188	6.4460	.0451	.6154	1.0743	1.5407
.0311	.5717	1.1114	6.0618	.0451	.6154	1.0743	1.5407

INTERMEDIATE VELOCITY DISTRIBUTION

Y	U	V	Y	U	V
7.028424	1.000000	0.000000	.913593	1.025205	.013297
3.532024	1.006643	.006752	.483282	1.072600	.029165
1.784255	1.007989	.011303	.139683	.814474	.354417

XB	MB	PB	DUDX	XB	MB	PB	DUDX
.0617	.6247	1.0662	.4506	.5472	.9010	.8192	-.3811
.0770	.6364	1.0561	.3804	.5622	.8901	.8289	-.4529
.0920	.6495	1.0447	.3211	.5772	.8748	.8426	-.5305
.1070	.6635	1.0324	.2812	.5920	.8556	.8598	-.6103
.1220	.6772	1.0203	.2502	.6070	.8332	.8800	-.6934
.1372	.6903	1.0087	.2202	.6220	.8082	.9026	-.7619
.1522	.7029	.9974	.1931	.6367	.7820	.9263	-.8084
.1672	.7151	.9865	.1682	.6517	.7556	.9501	-.8341
.1822	.7268	.9760	.1443	.6667	.7294	.9737	-.8415
.1972	.7381	.9659	.1218	.6815	.7039	.9965	-.8291
.2125	.7491	.9560	.1005	.6965	.6796	1.0182	-.8006
.2275	.7597	.9464	.0795	.7112	.6567	1.0384	-.7590
.2425	.7700	.9371	.0586	.7262	.6356	1.0568	-.7093
.2575	.7799	.9281	.0378	.7412	.6163	1.0735	-.6524
.2725	.7895	.9194	.0177	.7560	.5991	1.0883	-.5900
.2875	.7989	.9110	-.0014	.7710	.5840	1.1011	-.5264
.3025	.8081	.9027	-.0194	.7860	.5710	1.1119	-.4622
.3175	.8172	.8944	-.0365	.8007	.5602	1.1210	-.3979
.3325	.8262	.8863	-.0534	.8157	.5514	1.1282	-.3344
.3475	.8352	.8782	-.0701	.8307	.5447	1.1336	-.2697
.3625	.8442	.8701	-.0864	.8457	.5403	1.1372	-.2043
.3775	.8531	.8621	-.1029	.8607	.5383	1.1389	-.1377
.3925	.8619	.8542	-.1196	.8757	.5389	1.1384	-.0696
.4075	.8704	.8466	-.1363	.8905	.5423	1.1356	-.0002
.4225	.8786	.8393	-.1533	.9055	.5489	1.1302	.0738
.4375	.8861	.8325	-.1703	.9205	.5592	1.1218	.1522
.4525	.8929	.8265	-.1875	.9355	.5738	1.1096	.2365
.4675	.8984	.8215	-.2050	.9507	.5937	1.0928	.3304
.4825	.9023	.8181	-.2229	.9657	.6204	1.0700	.4343
.4975	.9041	.8165	-.2412	.9807	.6560	1.0390	.5534
.5022	.9041	.8165	-.2474	.9957	.7051	.9955	.6924
.5172	.9079	.8131	-.2515	1.0020	.7123	.9890	.7440
.5322	.9071	.8139	-.3137				

** AIRFOIL SOLUTION **

UPPER SURFACE

NC = 5

SHOCK LOC. = .485080, BETA = 90.000000, DELS = 0.000000, CDOQ = 11.000000, HO = .003000

FROM INITIAL CONDITIONS, NN = 6, X(INIT) = .053100

INTERMEDIATE VELOCITY DISTRIBUTION USING LAGRANGIAN FUNCTION

Y	U	V
6.971576	1.000000	0.000000
3.475218	1.006400	.006529
1.727478	1.007842	.010959
.856865	1.026044	.017403
.426659	1.083837	.183356
.051615	1.426226	.463849

X	MO	PO	DUDX	DOQ
.0621	1.1373	.6200	1.1337	-3.7580
.0711	1.1585	.6035	1.1038	-3.1607
.0800	1.1817	.5858	1.0809	-2.3741
.0890	1.2057	.5678	1.0647	-1.4564
.0980	1.2298	.5501	1.0570	-.4454
.1108	1.2634	.5260	1.0611	.5477
.1258	1.3004	.5003	1.0742	1.3344
.1408	1.3353	.4770	1.0996	2.2770
.1558	1.3677	.4561	1.1405	3.4859
.1706	1.3972	.4376	1.2001	4.8571
.1856	1.4237	.4215	1.2865	7.0045
.2006	1.4470	.4078	1.4200	10.9717
.2156	1.4670	.3963	1.6563	20.0125
.2306	1.4558	.4027	-.0144	-.3461
.2456	1.4446	.4092	-.0206	-.3391
.2606	1.4334	.4158	-.0266	-.3333
.2756	1.4220	.4225	-.0326	-.3346
.2906	1.4106	.4294	-.0385	-.3358
.3056	1.3991	.4364	-.0445	-.3369
.3203	1.3875	.4436	-.0503	-.3269
.3353	1.3758	.4509	-.0562	-.3342
.3503	1.3640	.4584	-.0620	-.3325
.3653	1.3520	.4661	-.0677	-.3289
.3803	1.3399	.4740	-.0733	-.3237
.3953	1.3276	.4821	-.0788	-.3182
.4103	1.3150	.4905	-.0842	-.3130
.4253	1.3022	.4991	-.0895	-.3103
.4403	1.2890	.5081	-.0946	-.3077
.4553	1.2756	.5175	-.0997	-.3051
.4706	1.2617	.5272	-.1048	-.3058
.4856	1.2475	.5373	-.1097	-.3073
.5156	.8136	.8977	-.0439	0.0000
.5456	.8112	.8998	-.0527	0.0000
.5756	.8069	.9037	-.0617	0.0000
.6056	.8008	.9093	-.0706	0.0000
.6361	.7927	.9165	-.0802	0.0000
.6661	.7827	.9256	-.0905	0.0000
.6961	.7708	.9363	-.1016	0.0000
.7261	.7570	.9488	-.1133	0.0000
.7566	.7414	.9629	-.1247	0.0000
.7866	.7244	.9782	-.1355	0.0000
.8166	.7062	.9945	-.1448	0.0000
.8471	.6871	1.0115	-.1525	0.0000
.8771	.6674	1.0290	-.1585	0.0000
.9071	.6471	1.0468	-.1622	0.0000
.9376	.6266	1.0646	-.1632	0.0000
.9676	.6060	1.0823	-.1603	0.0000
.9981	.5857	1.0996	-.1535	0.0000
1.0106	.5790	1.1052	-.1505	0.0000

DOWNSTREAM SOLUTION

NN = 3, H = .020000, RK * 5.000000

FROM UPPER SURFACE INTEGRATION, SHOCK LOC. * .4850, BETA * 90.0000, CS * 1.0000, CZ * 11.2041

X	MO	PO	P1
1.0800	.5775	1.1065	.9936
1.1600	.5759	1.1078	.9932
1.2400	.5742	1.1093	.9927
1.3200	.5724	1.1108	.9922
1.4000	.5704	1.1125	.9918
1.4800	.5683	1.1142	.9913
1.5600	.5662	1.1160	.9908
1.6400	.5639	1.1179	.9902
1.7200	.5615	1.1198	.9897
1.8000	.5591	1.1219	.9892
1.8800	.5565	1.1240	.9886
1.9600	.5538	1.1262	.9880
2.0400	.5510	1.1285	.9874
2.1200	.5481	1.1308	.9868
2.2000	.5452	1.1333	.9862
2.2800	.5421	1.1358	.9855
2.3600	.5389	1.1384	.9848
2.4400	.5356	1.1411	.9841
2.5200	.5322	1.1438	.9833
2.6000	.5287	1.1466	.9826
2.6800	.5251	1.1495	.9817
2.7600	.5214	1.1525	.9808
2.8400	.5176	1.1555	.9799
2.9200	.5136	1.1586	.9790
3.0000	.5096	1.1618	.9779
3.0800	.5055	1.1651	.9768
3.1600	.5012	1.1684	.9757
3.2400	.4968	1.1718	.9745
3.3200	.4924	1.1753	.9732
3.4000	.4878	1.1788	.9718
3.4800	.4831	1.1824	.9703
3.5600	.4783	1.1861	.9688
3.6400	.4733	1.1898	.9671
3.7200	.4683	1.1936	.9654
3.8000	.4632	1.1974	.9635
3.8800	.4580	1.2013	.9616
3.9600	.4527	1.2052	.9595
4.0400	.4473	1.2091	.9573
4.1200	.4418	1.2131	.9550
4.2000	.4363	1.2171	.9525
4.2800	.4307	1.2211	.9500
4.3600	.4250	1.2251	.9472
4.4400	.4192	1.2291	.9444
4.5200	.4135	1.2331	.9414
4.6000	.4076	1.2371	.9382
4.6800	.4018	1.2411	.9349
4.7600	.3959	1.2451	.9315
4.8400	.3900	1.2490	.9279
4.9200	.3841	1.2529	.9241
5.0000	.3782	1.2567	.9202
5.0800	.3723	1.2605	.9162
5.1600	.3665	1.2642	.9120
5.2400	.3607	1.2679	.9076
5.3200	.3549	1.2714	.9032
5.4000	.3493	1.2749	.8986
5.4800	.3437	1.2783	.8938
5.5600	.3382	1.2815	.8890

X	MO	PO	P1
5.6400	.3328	1.2847	.8840
5.7200	.3275	1.2878	.8790
5.8000	.3224	1.2907	.8739
5.8800	.3175	1.2935	.8688
5.9600	.3127	1.2962	.8636
6.0400	.3080	1.2988	.8584
6.1200	.3036	1.3012	.8532
6.2000	.2993	1.3035	.8481
6.2800	.2953	1.3056	.8430
6.3600	.2915	1.3077	.8381
6.4400	.2879	1.3095	.8334
6.5200	.2845	1.3113	.8289
6.6000	.2814	1.3129	.8247
6.6800	.2785	1.3143	.8208
6.7600	.2759	1.3157	.8172
6.8400	.2735	1.3169	.8141
6.9200	.2713	1.3179	.8114
7.0000	.2694	1.3189	.8092
7.0800	.2676	1.3197	.8074
7.1600	.2661	1.3205	.8062
7.2400	.2648	1.3211	.8053
7.3200	.2636	1.3217	.8049
7.4000	.2626	1.3222	.8047
7.4800	.2618	1.3226	.8049
7.5600	.2611	1.3229	.8052
7.6400	.2604	1.3232	.8056
7.7200	.2599	1.3234	.8061
7.8000	.2595	1.3237	.8065
7.8800	.2591	1.3238	.8069
7.9600	.2588	1.3240	.8071
8.0400	.2585	1.3241	.8072
8.1200	.2582	1.3242	.8071
8.2000	.2580	1.3244	.8069
8.2800	.2578	1.3244	.8064
8.3600	.2577	1.3245	.8057
8.4400	.2575	1.3246	.8049
8.5200	.2575	1.3246	.8038
8.6000	.2575	1.3246	.8027
8.6800	.2575	1.3246	.8015
8.7600	.2577	1.3245	.8003
8.8400	.2579	1.3244	.7991
8.9200	.2583	1.3242	.7981
9.0000	.2588	1.3240	.7973
9.0800	.2594	1.3237	.7969
9.1600	.2602	1.3233	.7969
9.2400	.2613	1.3228	.7974
9.3200	.2625	1.3222	.7984
9.4000	.2640	1.3215	.8000
9.4800	.2656	1.3207	.8022
9.5600	.2676	1.3198	.8050
9.6400	.2697	1.3187	.8083
9.7200	.2721	1.3175	.8120
9.8000	.2748	1.3162	.8162
9.8800	.2777	1.3148	.8206
9.9600	.2808	1.3132	.8254
10.0200	.2841	1.3115	.8303

*** PARTIAL PRESSURE DISTRIBUTION ***

UPPER SURFACE

X	PO	X	PO
.063470	.619965	.412470	.490487
.072470	.603469	.427470	.499146
.081470	.585753	.442470	.508124
.090470	.567763	.457470	.517454
.099470	.550092	.472470	.527171
.112470	.525961	.487470	.537318
.127470	.500328	.517470	.897671
.142470	.476999	.547470	.899811
.157470	.456071	.577470	.903693
.172470	.437612	.607470	.909277
.187470	.421536	.637470	.916546
.202470	.407762	.667470	.925562
.217470	.396282	.697470	.936342
.232470	.402678	.727470	.948833
.247470	.409169	.757470	.962851
.262470	.415782	.787470	.978176
.277470	.422528	.817470	.994468
.292470	.429411	.847470	1.011465
.307470	.436435	.877470	1.028967
.322470	.443606	.907470	1.046753
.337470	.450938	.937470	1.064631
.352470	.458437	.967470	1.082333
.367470	.466118	.997470	1.099616
.382470	.474002	1.007470	1.105244
.397470	.482115		

LOWER SURFACE

X	PO	X	PO
.018529	1.177701	.420090	.839263
.020009	1.168959	.435090	.832491
.021517	1.160234	.450090	.826406
.023051	1.151577	.465090	.821545
.024612	1.143043	.480090	.818078
.026199	1.134692	.495090	.816520
.027812	1.126590	.500090	.816503
.029449	1.118805	.515090	.813004
.031109	1.111406	.530090	.813069
.032792	1.104466	.545090	.819212
.034495	1.098058	.560090	.828932
.036219	1.092250	.575090	.842644
.037962	1.087110	.590090	.859821
.039721	1.082697	.605090	.879960
.041497	1.079062	.620090	.902554
.043287	1.076245	.635090	.926275
.045090	1.074271	.650090	.950098
.045090	1.074271	.665090	.973677
.060090	1.066244	.680090	.996525
.075090	1.056097	.695090	1.018198
.090090	1.044712	.710090	1.038355
.105090	1.032385	.725090	1.056828
.120090	1.020274	.740090	1.073507
.135090	1.008660	.755090	1.088273
.150090	.997403	.770090	1.101075
.165090	.986527	.785090	1.111949
.180090	.976032	.800090	1.120956
.195090	.965859	.815090	1.128210
.210090	.955970	.830090	1.133649
.225090	.946373	.845090	1.137233
.240090	.937082	.860090	1.138864
.255090	.928106	.875090	1.138398
.270090	.919421	.890090	1.135687
.285090	.910968	.905090	1.130228
.300090	.902668	.920090	1.121767
.315090	.894449	.935090	1.109628
.330090	.886292	.950090	1.092028
.345090	.878183	.965090	1.070024
.360090	.870108	.980090	1.038968
.375090	.862097	.995090	.995450
.390090	.854221	1.000090	.989026
.405090	.846568		

APPENDIX F

MIR SUBROUTINES LISTING

```
      SUBROUTINE IOUPSTM
C
C   THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C   OUTPUT FOR SUBROUTINE UPSTRM
C
      COMMON/YUVSAV/    NNINIT,NNSPR,NNDWN
     1       ,YI(10,2),UI(10,2),VI(10,2),YUV(96)
      COMMON/AINPUT/       ATN(24),NN(7)   ,H(6)
      COMMON/OUTCOM/
     1    AX(160)   ,ARMO(160),AY(10)    ,ARM(10)   ,DUM(140) ,II,II2
      DIMENSION ISTAR(5),ITITLE(2)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,2)/10H UPSTREAM ,10HSOLUTION */
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5)
     1   ,NN(1),NN(2),ATN(1),(AIN(I),I=8,10),H(1),AIN(11),AIN(12)
      IF(H(1).GT.0.008)GO TO 5
      WRITE(6,340)
      RETURN
    5 CALL UPSTRM
      IF(II.EQ.0) RETURN
      IHALF = II/2
      J = MOD(II,2)
      IHALF1 = IHALF
      IF(J.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 15
      DO 10 I=1,IHALF
   10 WRITE(6,300) AX(I),ARMO(I),AX(I+IHALF1),ARMO(I+IHALF1)
      IF(J.NE.1) GO TO 18
   15 WRITE(6,300) AX(IHALF1),ARMO(IHALF1)
   18 IF(II2.EQ.0) WRITE(6,260)
      IHALF = II2/2
      J = MOD(II2,2)
      IHALF1 = IHALF
      IF(J.EQ.1) IHALF1=IHALF+1
      WRITE(6,410)
      IF(II2.EQ.1) GO TO 25
      DO 20 I=1,IHALF
   20 WRITE(6,300) AY(I),ARM(I),AY(I+IHALF1),ARM(I+IHALF1)
      IF(J.NE.1) GO TO 28
   25 WRITE(6,300) AY(IHALF1),ARM(IHALF1)
   28 WRITE(6,320)  AIN(20),AIN(21)
      DO 30 I=1,NNINIT
   30 WRITE(6,330) YI(I,1),UI(I,1),VI(I,1),YI(I,2),UI(I,2),VI(I,2)
      RETURN
  200 FORMAT(1H1,4(/)7X,12A10,
     1            ///20X,4HNN =,I2,6H, NA =,I2//20X,9HDV00(I) =,
     2 F10.4,9H,   X00 =  F8.4,                   9H,    RMT =  F8.4,
     3 9H,   CDY =   F8.4,7H,    H =   F8.4//20X,13HYINF(UPPER) =F8.4,
     4  17H,    YINF(LOWER) =F8.4)
  260 FORMAT(/20X,39H**********INTEGRATION WAS NOT COMPLETED )
  300 FORMAT(30X,2(10X,2F10.6))
  320 FORMAT(//47X,4HDE =,F10.6,10X,5HYSO =,F10.6///52X,25H INITIAL VELO
     1CITY PROFILE//39X,13HUPPER SURFACE,27X,13HLOWER SURFACE//17X,
     22(19X,1HY,9X,1HU,9X,1HV))
  330 FORMAT(20X,2(10X,3F10.6))
  340 FORMAT(//20X,29H**********STEP SIZE TOO SMALL)
  400 FORMAT(//27X,2(18X,1HX,9X,2HMO))
  410 FORMAT(//26X,2(19X,1HY, 9X,1HM))
      END
```

```
      SUBROUTINE UPSTRM
C
C     THIS SUBROUTINE CALCULATES THE UPSTREAM FLOW CONDITIONS
C
      COMMON        C        ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X          ,XA      ,VN      ,VS      ,H       ,DY
     1   ,YO     ,PO         ,RO      ,UO      ,VO      ,RMO     ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/BCOM/            XO      ,DVOO    ,L
      COMMON/ECOM/UOSAV  ,ROSAV  ,XSAV    ,DVOOF   ,XSTG(11),YSTG(11)
     1   ,DUM(22),XAF(50),YAF(50,2)
      COMMON/AINPUT/          DVOOI   ,DDUM(6),XOO     ,RMT     ,CDY
     1   ,YU     ,YL         ,EDUM(7),DE      ,YSO     ,YS      ,FDUM(2)
     2   ,NNI    ,NA         ,NDUM(5),HDWN    ,HDUM(5)
      COMMON/OUTCOM/
     1     AX(160)  ,ARMO(160),AY(10)    ,ARM(10)   ,CDUM(140) ,II,NNO
      COMMON/YUVSAV/    NNINIT,NNSPR,NNDWN
     1        ,YI(10,2),UI(10,2),VI(10,2),YUV(96)
      DIMENSION BX(6),BYO(6),BMO(6),Y1(2),Y2(2),A1U(2),A2U(2),A1V(2),
     1     A2V(2)
C
C     INITIALIZE INPUT
      G(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
      NTERM = 3
      CSO = FM-.08
      CS1 = CSO-0.2
      NN = NNI
      DVOC = DVOOI
      H = HDWN
      Y(1,1) = YU
      Y(2,1) = YL
      II = 0
      NNO = 0
      RS =  (1./(7.*CK) + 1.)**2.5
      L = 4
      XO = XOO
      X = 0.0
C
C     INITIALIZE FLOW TO FREESTREAM CONDITIONS
      DO 10 J=1,2
      DO   5 N = 2, NN
      Y(J,N) = Y(J,N-1)*0.5
      P(J,N) = 1.
      R(J,N) = 1.
      U(J,N) = 1.
      V(J,N) = 0.
      RM(J,N) = FM
    5 DU(J,N) = 0.0
      RM(J,1) = FM
      U(J,1) = 1.0
   10 V(J,1) = 0.0
      YO = 0.
      RO = 1.
      PO = 1.
      UO = 1.
      VS = 1.
      VO = 0.
      RMO = FM
C
      II = II+1
      AX(II) = X
      ARMO(II) = RMO
C
      DVA = 0.0
      DO 26 K=1,100
      DO 24 KK=1,NTERM
C
C     PERFORM A FLOW INTEGRATION STEP USING SUBROUTINE DIST
      ISGN = 1
```

```
         DO 22 J=1,2
         CALL OUNS(ISGN,J)
         DO 21 N = 3, NN
      21 CALL INAS(ISGN,J,N,1)
      22 ISGN = -1
         IF(K.LE.3) GO TO 23
         CALL DIST(-1,1,NN,DYA,DUA,DVA)
      23 CALL DIST(1,2,NN,DYO,DVS,DVO)
C
         X = X + H
         VS = VS + H*DVS
         VO = VO + H*(DVO - DVA)
         IF(VS.LT.VO)RETURN
         UO = SQRT(VS**2 - VO**2)
         YO = YO + H*DYO
         RO = ((C - VS**2        )/(C - 1.))**2.5
         PO = RO**1.4
         RMO = VS*SQRT(PO/(1.4*CK*PO))
C
         IF(RMO.LE.RMT) GO TO 25
C
      24 CONTINUE
C      CALCULATE CERTAIN FACTORS USED IN DETERMINING RBUB
      25 II = II+1
         AX(II) = X
         ARMO(II) = RMO
C
         IF (RMO .LE. RMT) GO TO 27
C
      26 CONTINUE
C
      27 I = 0
C
      28 M = ABS((5.0-X)/H)
C
C      DEPENDING ON DYO (THE SLOPE OF THE STAGNATION STREAMLINE) PERFORM
C      A FLOW INTEGRATION STEP USING SUBROUTINE STMR OR LUMR
         IF (DYO .GE. CDY  .OR. DY .GE. CDY) GO TO 35
         DO 34 K = 1, M
C
         DO 31 KK=1,NTERM
C
C      PERFORM A FLOW INTEGRATION STEP USING LUMR
         ISGN = 1
         DO 30 J=1,2
         CALL OUNS(ISGN,J)
         DO 29 N = 3, NN
      29 CALL INAS(ISGN,J,N,1)
      30 ISGN = -1
         CALL LUMR(-1,1,NN,DYA,DUA,DVA)
         CALL LUMR(1,2,NN,DYO,DVS,DVO)
C
         X = X + H
         VS = VS + H*DVS
         VO = VO + H*(DVO - DVA)
         UO = SQRT(VS**2 - VO**2)
         YO = YO + H*DYO
         PO = ((C - VS**2        )/(C - 1.))**2.5
         PO = RO**1.4
         RMO = VS*SQRT(RO/(1.4*CK*PO))

         IF (RMO .LE. CSO) GO TO 42

         IF(DYO.GE.CDY) GO TO 32

      31 CONTINUE

      32 II = II+1
         AX(II) = X
         ARMO(II) = RMO
```

```
      IF (DVO .GE. CDY) GO TO 35
   34 CONTINUE
   35 T = ATAN(VO/UO)
      DO 40 K = 1, M
      DO 38 KK=1,NTERM
C
C   PERFORM A FLOW INTEGRATION STEP USING STMR
      ISGN = 1
      DO 37 J=1,2
      CALL OUNS(ISGN,J)
      DO 36 N = 3, NN
   36 CALL INAS(ISGN,J,N,1)
   37 ISGN = -1
      CALL STMR(NN,T,DY,DVS)
C
C
C   IF RMO (MACH NUMBER AT THE STAGNATION STREAMLINE) IS CONSIDERABLY
C   LESSENED, SAVE FLOW PARAMETERS AT THIS STEP
      IF(RMO.LE.CSO) GO TO 42
   38 CONTINUE
      II = II+1
      AX(II) = X
      ARMO(II) = RMO
C
      IF(X.GE.7..OR.X.LT.0.0.OR.DVS.GE.0.0) RETURN
   40 CONTINUE
      RETURN
C
   42 II = II+1
      AX(II) = X
      ARMO(II) = RMO
C
      IF(CS1.LT.CSO) GO TO 43
C
C   SAVE FLOW PROPERTIES AT THIS STEP
      I = I + 1
      BX(I) = X
      BYO(I) = YO
      BMO(I) = RMO
      IF (I .EQ. 3) H = 0.002
      CSO = CSO - 0.05
      IF(I-4) 28,60,60
C
C   CS1 IS LESS THAN CSO FOR THE FIRST PASS THROUGH THE LOOP, SO
C   STATEMENT 43 IS EXECUTED ONLY ONCE
   43 DO 44 J=1,2
      Y1(J) = Y(J,NN)-YO
      Y2(J) = Y(J,NN-1)-YO
      CALL A1SUP(Y1(J),Y2(J),UO,U(J,NN),U(J,NN-1),A1U(J))
      CALL A2SUP(Y1(J),Y2(J),UO,U(J,NN),U(J,NN-1),A2U(J))
      CALL A1SU3(Y1(J),Y2(J),VO,V(J,NN),V(J,NN-1),A1V(J))
      CALL A2SU3(Y1(J),Y2(J),VO,V(J,NN),V(J,NN-1),A2V(J))
      VO = -VO
   44 YO = -YO
C
C   ADD NA STRIPS TO THE FLOW INTEGRATION PROCESS AT THIS POINT
      N1 = NN + 1
      NN = NN + NA
      DO 47 J=1,2
      DO 46 N = N1, NN
      Y(J,N) = (Y(J,N-1)-YO)/2.+YO
      U(J,N) = G(UO,A1U(J),A2U(J),Y(J,N)-YO)
      V(J,N) = G(VO,A1V(J),A2V(J),Y(J,N)-YO)
      VSQ = U(J,N)*U(J,N)+V(J,N)*V(J,N)
      R(J,N) = ((C-VSQ)/(C-1.))**2.5
      P(J,N) = R(J,N)**1.4
```

```
   46 RM(J,N) = SQRT(VSQ*R(J,N)/(1.4*CK*P(J,N)))
      VO = -VO
   47 YO = -YO
C
      H = H/5.
      IF (H .LT. 0.005) H = 0.005
      CSO = CS1
      GO TO28
C
C   SAVE THE Y STATIONS AND THE MACH NUMBERS AT THE FINAL INTEGRATION
C   STEP
   60 DO 61 J=1,NN
      AY(J) = Y(1,J)
   61 ARM(J) = RM(1,J)
      NNO = NN+1
      AY(NNO) = YO
      ARM(NNO) = RMO
      XO = X
C
C   EXTRAPOLATE RMO TO A VALUE OF ZERO USING THE OUTPUT OF THE FOUR
C   PREVIOUSLY COMPUTED STEPS
      DO 63 K = 1, 200
      X = X + 0.0005
      CALL LGRNGN(BMO(1),BMO(2),BMO(3),BMO(4),
     1BX(1),BX(2),BX(3),BX(4),X,RMO)
      IF(RMO.LE.0.0) GO TO 64
   63 CONTINUE
C
C   DE IS THE CALCULATED VALUE FROM THE LAST INTEGRATION STEP TO THE
C   STAGNATION POINT
   64 DE = X - XO
C
C   EXTRAPOLATE YSO FROM THE OUTPUT OF FOUR PREVIOUSLY COMPUTED STEPS
C   AND THE CALCULATED VALUE OF X
      CALL LGRNGN(BYO(1),BYO(2),BYO(3),BYO(4),
     1BX(1),BX(2),BX(3),BX(4),X,YSO)
      UOSAV = UO
      ROSAV = RO
      XSAV = XO
      XSTG(1) = -DE
      XSTAG = XO
      YSTG(1) = YO
      DXSTAG = DE/10.
C
C   CALCULATE TEN COORDINATES ALONG THE STAGNATION STREAMLINE
      DO 70 I=2,11
      XSTAG = XSTAG+DXSTAG
      CALL LGRNGN(BYO(1),BYO(2),BYO(3),BYO(4),BX(1),BX(2),BX(3),BX(4),
     1XSTAG,YSTG(I))
   70 XSTG(I)  = XSTG(I-1) +DXSTAG
C
C   CALCULATE FIFTY POINTS ALONG THE FIRST 3 PER CENT OF THE AIRFOIL
C   NOSE
      DO 76 J=1,2
      XAF(1) = 0.
      YAF(1,J) = 0.
      DO 75 I=2,50
      XAF(I) = XAF(I-1) +.0006
      CALL ARFL(XAF(I),ADUM,YAF(I,J),BDUM,CDUM,J)
      IF(J.EQ.2) YAF(I,2) = -YAF(I,2)
   75 CONTINUE
   76 CONTINUE
C
C   SAVE OUTPUT FLOW PARAMETERS OF THIS SUBROUTINE
      DO 80 J=1,2
      DO 78 I=1,NN
      YI(I,J) = Y(J,I)
      UI(I,J) = U(J,I)
   78 VI(I,J) = V(J,I)
      NNINIT = NNO
```

```
      YI(NNO,J) = YO
      UI(NNO,J) = UO
      VI(NNO,J) = VO
      YO = -YO
   80 VO = -VO
C
      RETURN
      END
```

```
      SUBROUTINE IOSTGNA
C
C   THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C   OUTPUT FOR SUBROUTINE STAGNA
C
      COMMON/ECOM/DUM(26),XSTAG(11),YSTAG(11),XARFL(50),YARFL(50,2)
      COMMON/AINPUT/       AIN(24),NM(7)   ,H(6)
      DIMENSION ISTAR(5),ITITLE(2)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,2)/10HSTAGNATION,10H SOLUTION*/
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5)
      WRITE(6,210) AIN(2)
      WRITE(6,220) AIN(20),AIN(21)
      CALL STAGNA
      WRITE(6,410)
      DO 10 I=1,25
   10 WRITE(6,400) XARFL(I)    ,YARFL(I,1),YARFL(I,2),XARFL(I+25),
     1  YARFL(I+25,1),YARFL(I+25,2)
      WRITE(6,430)
      DO 20 I=1,5
   20 WRITE(6,420) XSTAG(I),YSTAG(I),XSTAG(I+6),YSTAG(I+6)
      WRITE(6,420) XSTAG(6),YSTAG(6)
      WRITE(6,370) AIN(22),DUM(4)
      RETURN
  200 FORMAT(1H1,4(/)7X,12A10)
  210 FORMAT(//20X,4HXS =,F10.6)
  220 FORMAT(/20X,34HFROM THE UPSTREAM SOLUTION,    DE =,F10.6,
     1 9H,    YSD = ,F10.6)
  370 FORMAT(/ 46X,4HYS =,F10.6,10X,9HDVOO(F) =,F10.6)
  400 FORMAT(13X,2(10X,3F12.6))
  410 FORMAT(///57X,19HAIRFOIL COORDINATES//13X,2(18X,1HX,7X,8HY(UPPER),
     1  4X,8HY(LOWER)))
  420 FORMAT(   25X,2(10X,2F12.6))
  430 FORMAT(///56X,21HSTAGNATION STREAMLINE//21X,2(21X,1HX,11X,1HY))
      END
```

```
      SUBROUTINE STAGNA
C
C   THIS SUBROUTINE CALCULATES THE STAGNATION STREAMLINE GEOMETRY AND
C   CERTAIN AIRFOIL COORDINATES
C
      COMMON       C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/BCOM/         XO       ,DVOO    ,L
      COMMON/ECOM/UO       ,RO      ,X       ,DVOOF   ,XIN(11),YIN(11)
     1    ,XOU(11),YOU(11),XAF(50),YAF(50,2)
      COMMON/AINPUT/       DVOOI    ,XS      ,DUM(17),DE      ,YSO
     1    ,YS      ,DDUM(2),NDUM(7),HDUM(6)
C
C   COMPUTE DVOOF FOR SELECTED STAGNATION POINT
      CALL ARFL(XS    ,ADUM,YS,DYS,DDYS,2)
      RA =1./ ABS(DDYS/(1.+DYS**2)**1.5)
      DVOOF = (2./DE+2./(DE+RA)-1./RA)*UO/
     1    (RS/RO+(X/(X+DE))**L*(1.+DE/RA))
C
C   ADJUST COORDINATES FOR AIRFOIL NOSE AND STAGNATION STREAMLINE
C   GEOMETRY TO ONE CARTESIAN FRAME
      DO 20 I=1,11
      XOU(I) = XIN(I)+XS
   20 YOU(I) = YIN(I)-YS-YSO
      RETURN
      END
```

```
      SUBROUTINE IOLWRCT
C
C   THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C   OUTPUT FOR SUBROUTINE LWRCRIT
C
      COMMON/AINPUT/      AIN(24),NN1(7) ,HI(6)
      COMMON/YUVSAV/NN0,NN2,NN3,YUV(156)
      COMMON/OUTCOM/
     1    AXB(160) ,ARMB(160),DUM(160) ,II        ,II2
      DIMENSION ISTAR(5),ITITLE(4)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,4)/10HTEST OF CR,10HITICALITY*,10H   LOWER S,
     1   10HURFACE     /
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5),
     1  (ITITLE(I),I=3,4)
      WRITE(6,220) NN0,AIN(20)
      WRITE(6,230) AIN(22),AIN(21)
      CALL LWRCRIT
      IF(II.EQ.0) RETURN
      IHALF = II/2
      J = MOD(II,2)
      IHALF1 = IHALF
      IF(J.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 45
      DO 40 I=1,IHALF
   40 WRITE(6,300) AXB(I),ARMB(I),AXB(I+IHALF1),ARMB(I+IHALF1)
      IF(J.NE.1) GO TO 48
   45 WRITE(6,300) AXB(IHALF1),ARMB(IHALF1)
   48 RETURN
  200 FORMAT(1H1,4(/),7X,12A10/57X,2A10/)
  220 FORMAT(/20X,34HFROM THE UPSTREAM SOLUTION,    NN =,I2,8H,    DE =,
     1    F10.6)
  230 FORMAT(/20X,36HFROM THE STAGNATION SOLUTION,    YS =,F10.6,
     1  9H,    YSO =,F10.6)
  300 FORMAT(30X,2(10X,2F10.4))
  400 FORMAT(//29X,2(18X,2HXB,8X,2HMB))
      END
```

```
      SUBROUTINE LWRCRIT
C
C     THIS SUBROUTINE CALCULATES MACH NUMBER FOR A SELECTED NUMBER OF
C     POINTS ON THE LOWER SURFACE
C
      COMMON       C        ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X         ,XA      ,VN      ,VS      ,H       ,DY
     1    ,YO      ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/AINPUT/     DYOOI   ,XAS     ,DUM(17),DE      ,YSO
     1    ,YS      ,DDUM(2),NN1(7)  ,HI(6)
      COMMON/YUVSAV/     NNO,NNSPR,NNDWN
     1         ,YI(10,2),UI(10,2),VI(10,2),YUV(96)
      COMMON/OUTCOM/
     1     AXB(160) ,ARMB(160),ADUM(160),II        ,II2
      D(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
C
C     INITIALIZE INPUT
      CYD = DUM(4)
      DO 10 N=2,NNO
      Y(2,N) = YI(N,2)+YSO+YS
      U(2,N) = UI(N,2)
      V(2,N) = VI(N,2)
   10 R(2,N) = ((C-U(2,N)*U(2,N)-V(2,N)*V(2,N))/(C-1.))**2.5
C
C     CALCULATE FACTORS USED IN DETERMINING RBUB
      Y1 = Y(2,NNO-1)-YS-YSO-YI(NNO,2)
      Y2 = Y(2,NNO-2)-YS-YSO-YI(NNO,2)
      CALL A1SUB(Y1,Y2,U(2,NNO),U(2,NNO-1),U(2,NNO-2),A1U)
      CALL A2SUB(Y1,Y2,U(2,NNO),U(2,NNO-1),U(2,NNO-2),A2U)
      CALL A1SUB(Y1,Y2,V(2,NNO),V(2,NNO-1),V(2,NNO-2),A1V)
      CALL A2SUB(Y1,Y2,V(2,NNO),V(2,NNO-1),V(2,NNO-2),A2V)
      BO = U(2,NNO)*R(2,NNO)
      B1 = U(2,NNO-1)*R(2,NNO-1)
      B2 = U(2,NNO-2)*R(2,NNO-2)
      CALL A1SUB(Y1,Y2,BO,B1,B2,A1C)
      CALL A2SUB(Y1,Y2,BO,B1,B2,A2C)
      CALL ARFL (XAS, XBS, YS, DYBS, DDYBS,2)
      DBS = (DE + XBS)*SQRT(1. + DYBS**2)/DYBS
      YDS = (DE + XBS)/DYBS - YSO - YI(NNO,2)
      UO = D(U(2,NNO),A1U,A2U,YDS)
      VO = D(V(2,NNO),A1V,A2V,YDS)
      VSQ = UO*UO +VO*VO
      IF((C-VSQ).LE.0) RETURN
      RO = ((C-VSQ)/(C-1.))**2.5
      VS  = (UO +VO*DYBS)/SQRT(1. +DYBS**2)
      CLA= DBS *RO*VS/2.
      CLB = BO*YDS + A1C*YDS**2/2. + A2C*YDS**3/3.
C
      DX = .003
   40 XA = 0.
      II = 0
C
      DO 80  I = 1,20
C
C     DETERMINE RBUB FOR THIS XA
      XA = XA+DX
      CALL ARFL (XA, XB, YB, DYB, DDYB,2)
      YO = (DE + XB)/DYB + YB
      YD = YO-YSO-YS-YI(NNO,2)
C
C     LET Y2 = Y(2,NNO-4)
      J=3
      Y1 = Y(2,NNO-J)-YS-YSO-YI(NNO,2)
      Y2 = Y(2,NNO-J-1)-YS-YSO-YI(NNO,2)
      CALL A1SUB(Y1,Y2,U(2,NNO),U(2,NNO-J),U(2,NNO-J-1),A1U)
      CALL A2SUB(Y1,Y2,U(2,NNO),U(2,NNO-J),U(2,NNO-J-1),A2U)
      CALL A1SUB(Y1,Y2,V(2,NNO),V(2,NNO-J),V(2,NNO-J-1),A1V)
      CALL A2SUB(Y1,Y2,V(2,NNO),V(2,NNO-J),V(2,NNO-J-1),A2V)
```

```
      B1 = U(2,NNO-J)*R(2,NNO-J)
      B2 = U(2,NNO-J-1)*R(2,NNO-J-1)
      CALL A1SUB(Y1,Y2,B0,B1,B2,A1C)
      CALL A2SUB(Y1,Y2,B0,B1,B2,A2C)
   54 U0 = D(U(2,NNO),A1U,A2U,YD)
      V0 = D(V(2,NNO),A1V,A2V,YD)
      R0 = ((C-U0*U0-V0*V0)/(C-1.))**2.5
      CT = 1./SQRT(1. + DYB**2)
      ST = DYB*CT
      DB = (DE + XB)/ST
      VS = U0*CT + V0*ST
      CL = B0*YD + A1C*YD**2/2. + A2C*YD**3/3. + CLA - CLB
      RBUB = 6.*CL/DB-VS*R0-4.*VS*R0*CYD
C
      IF(RBUB.LT.0.0) GO TO 80
C
      UBS = 0.1
      RBUBP = RBUB**0.4*(C-1.)
C
C    USING NEWTON RAPHSONG METHOD ITERATE ON UB UNTIL RUP = RBUBP
      UB=0.1
      DO 60 K=1,50
      RUP = C*UB**0.4-UB**2.4
      IF(ABS(RUP-RBUBP).LT..000001) GO TO 70
      DRUPDU = 0.4*C/UB**0.6-2.4*UB**1.4
      UB = UB+(RBUBP-RUP)/DRUPDU
      IF(UB.LT.0.) UBS=UBS+.05
      IF(UB.LT.0.) UB=UBS
      IF(UBS.GT.1.) GO TO 80
   60 CONTINUE
C
      IF(I.GT.4.OR.DX.LT.0.0004) GO TO 90
      DX = .0003
      GO TO 40
C
   70 RB = ((C - UB**2)/(C - 1.))**2.5
      PB = RB**1.4
C
C    CALCULATE RMB FOR THIS UB
      RMB =  UB/SQRT(1.4*CK*PB/RB)
C
      II = II+1
      ARMB(II) = RMB
      AXB(II) = XB
C
   80 CONTINUE
C
   90 RETURN
      END
```

```
      SUBROUTINE IOLWRIN(ICRIT,L)
C
C    THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C    OUTPUT FROM SUBROUTINE LWRINIT
C
      COMMON/AINPUT/      AIN(24),NN1(7) ,HI(6)
      COMMON/YUVSAV/   NNINIT,NNSPR,NNDWN,YUV(156)
      COMMON/OUTCOM/
     1    AXA(160) ,ARMO(160),ADU(160) ,II       ,II2
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      COMMON/RBUBCM/RBUB      ,UBINIT    ,IRBUB
      DIMENSION ISTAR(5),ITITLE(4)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,4)/10H* AIRFOIL ,10HSOLUTION *,10H   LOWER S,
     1 10HURFACE     /
      J=2
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5),
     1  (ITITLE(I),I=3,4)
      WRITE(6,210) NN1(4),AIN(5),AIN(6),AIN(13),HI(3),HI(4)
      M =  (1.0-AIN(13))/HI(4)+(AIN(13)-AIN(5))/HI(3)
      NN1(7) = 0
      IF(M.LT.470.OR.ICRIT.EQ.2.OR.NNINIT.GE.NN1(4)) GO TO 4
      IF(ICRIT.NE.2) WRITE(6,360)
      IF(M.GT.470) WRITE(6,370)
      IF(NNINIT.LT.NN1(4)) WRITE(5,380)
      RETURN
    4 CALL LWRINIT(ICRIT)
      IF(IRBUB.EQ.0) WRITE(6,330) RBUB,UBINIT
      IF(IRBUB.EQ.1) WRITE(6,350) RBUB
      IF(IRBUB.EQ.2) WRITE(6,340) RBUB
      IF(II.EQ.0) RETURN
      IHALF = II/2
      K = MOD(II,2)
      IHALF1 = IHALF
      IF(K.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 15
      DO 10 I=1,IHALF
   10 WRITE(6,300) AXA(I),ARMO(I),PP(I,J),ADU(I),AXA(I+IHALF1),
     1    ARMO(I+IHALF1),PP(I+IHALF1,J),ADU(I+IHALF1)
      IF(K.NE.1) GO TO 18
   15 WRITE(6,300) AXA(IHALF1),ARMO(IHALF1),PP(IHALF1,J),ADU(IHALF1)
   18 IF(NN1(7).EQ.0) GO TO 28
      CALL INVELOC(L,J)
      CALL SUBCRT2(J)
      NN17P1 = NN1(7)+1
      II3 = II-NN1(7)
      IF(II3.LE.0) GO TO 28
      IHALF = II3/2
      IMIDL = IHALF+NN1(7)
      K = MOD(II3,2)
      IHALF1 = IHALF
      IF(K.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II3.EQ.1) GO TO 25
      DO 20 I=NN17P1,IMIDL
   20 WRITE(6,300) AXA(I),ARMO(I),PP(I,J),ADU(I),AXA(I+IHALF1),
     1  ARMO(I+IHALF1),PP(I+IHALF1,J),ADU(I+IHALF1)
      IF(K.NE.1) GO TO 28
   25 WRITE(6,300) AXA(IMIDL+1),ARMO(IMIDL+1),PP(IMIDL+1,J),ADU(IMIDL+1)
   28 IF(II3.EQ.1) WRITE(6,260)
      RETURN
  200 FORMAT(4(/),7X,12A10/57X,2A10)
  210 FORMAT(         //20X,4HNB =,I2//20X,4HXA =,F8.4,9H,   CYD =
     1 F8.4, 8H,   CX =F8.4, 9H,   HSO =,F8.4, 8H,   HO =F8.4)
  260 FORMAT(/20X,39H**********INTEGRATION WAS NOT COMPLETED )
  300 FORMAT(10X,2(10X,4F10.4))
  310 FORMAT(20X,4A10,10X,4F10.4)
  320 FORMAT(20X,4F10.4,10X,4A10)
```

```
  330 FORMAT(/ 47X,6HRBUB =,F10.6,10X,4HUB =,F10.6)
  340 FORMAT(/ 39X,6HRBUB =,F10.6,40H******FLOW CONDITIONS CANNOT BE MAT
     1CHED    )
  350 FORMAT(//59X,6HRBUB =,F10.6)
  360 FORMAT(//20X,62H**********SUPERCRITICAL FLOW IS NOT PERMITTED ON L
     1OWER SURFACE    )
  370 FORMAT(//20X,29H**********STEP SIZE TOO SMALL )
  380 FORMAT(//20X,49H**********INSUFFICIENT NUMBER OF STRIPS AVAILABLE)
  400 FORMAT(//10X,2(17X,2HXB,8X,2HMB,8X,2HPB,7X,4HDUDX))
      END
```

```
      SUBROUTINE LWRINIT(ICRIT)
C
C   THIS SUBROUTINE CALCULATES THE INITIAL CONDITIONS USED IN THE
C   LOWER SURFACE
C
      COMMON      C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X       ,XA      ,VN      ,VS      ,H       ,DY
     1    ,YO     ,PO     ,RO      ,UO      ,VO      ,RMO     ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/CCOM/XB       ,YB      ,DYB     ,DDYB    ,DUB     ,PB
     1    ,UB     ,RMB     ,DB      ,HS      ,CPA
      COMMON/DCOM/CS       ,CZ      ,DV1     ,Q1      ,DQ1     ,RK
     1    ,VOO    ,ISKIP
      COMMON/AINPUT/       DVOOI   ,XAS     ,BDUM(2),XAI     ,CYD
     1    ,CDUM(6),CXI     ,DDUM(6),DE      ,YSO     ,YS      ,EDUM(2)
     2    ,NN1(3) ,NNLWR   ,NN2(3) ,H1(2)   ,HSO     ,H2(3)
      COMMON/YUVSAV/    NNO,NNSPR,NNDWN
     1         ,YI(10,2),UI(10,2),VI(10,2),YUV(96)
      COMMON/COMNN/NN
      COMMON/RBUBCM/RBUB      ,UBINIT   ,IRBUB
      COMMON/OUTCOM/ADUM(480),II        ,II2
      D(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
C
C   INITIALIZE INPUT
      DCX = .004
      II2=0
      II = 0
      NN1(7) = 0
      IF((XAI-CXI + 3.*DCX).GT.0.0) RETURN
      NN = NNLWR
      XA = XAI
      X = -DE
      HS = HSO
      DO 45 N = 1, NNO
      Y(2,N) = YI(N,2)+YSO+YS
      U(2,N) = UI(N,2)
      V(2,N) = VI(N,2)
      VSQ = U(2,N) *U(2,N) +V(2,N)*V(2,N)
      R(2,N) = ((C-VSQ)/(C-1.))**2.5
      P(2,N) = R(2,N)**1.4
   45 RM(2,N) = SQRT(VSQ*R(2,N)/(1.4*CK*P(2,N)))
C
C   CALCULATE FACTORS USED IN DETERMINING RBUB
      Y1 = Y(2,NNO-1)-YS-YSO-YI(NNO,2)
      Y2 = Y(2,NNO-2)-YS-YSO-YI(NNO,2)
      CALL A1SUB(Y1,Y2,U(2,NNO),U(2,NNO-1),U(2,NNO-2),A1U)
      CALL A2SUB(Y1,Y2,U(2,NNO),U(2,NNO-1),U(2,NNO-2),A2U)
      CALL A1SUB(Y1,Y2,V(2,NNO),V(2,NNO-1),V(2,NNO-2),A1V)
      CALL A2SUB(Y1,Y2,V(2,NNO),V(2,NNO-1),V(2,NNO-2),A2V)
      BO = U(2,NNO)*R(2,NNO)
      B1 = U(2,NNO-1)*R(2,NNO-1)
      B2 = U(2,NNO-2)*R(2,NNO-2)
      CALL A1SUB(Y1,Y2,BO,B1,B2,A1C)
      CALL A2SUB(Y1,Y2,BO,B1,B2,A2C)
      CALL AREL (XAS, XBS, YS, DYBS, DDYBS,2)
      DBS = (DE + XBS)*SQRT(1. + DYBS**2)/DYBS
      YDS = (DE+XBS)/DYBS-YSO-YI(NNO,2)
      UO = D(U(2,NNO),A1U,A2U,YDS)
      VO = D(V(2,NNO),A1V,A2V,YDS)
      RO = ((C - UO**2 - VO**2)/(C - 1.))**2.5
      VS  = (UO +VO*DYBS)/SQRT(1. +DYBS**2)
      CLA= DBS *RO*VS/2.
      CLB = BO*YDS + A1C*YDS**2/2. + A2C*YDS**3/3.
      CALL AREL (XA, XB, YB, DYB, DDYB,2)
      YO = (DE + XB)/DYB + YB
      YD = YO-YSO-YS-YI(NNO,2)
C
C   FIND A VALUE FOR Y2 SUCH THAT YD IS LESS THAN Y2
      J=0
```

```
   48 J = J+1
      Y1 = Y(2,NNO-J)-YS-YSO-YI(NNO,2)
      Y2 = Y(2,NNO-J-1)-YS-YSO-YI(NNO,2)
      IF(YD-Y2) 52,48,48
C
C     CALCULATE RBUB
   52 IF(J.EQ.1) GO TO 54
      CALL A1SUB(Y1,Y2,U(2,NNO),U(2,NNO-J),U(2,NNO-J-1),A1U)
      CALL A2SUB(Y1,Y2,U(2,NNO),U(2,NNO-J),U(2,NNO-J-1),A2U)
      CALL A1SUB(Y1,Y2,V(2,NNO),V(2,NNO-J),V(2,NNO-J-1),A1V)
      CALL A2SUB(Y1,Y2,V(2,NNO),V(2,NNO-J),V(2,NNO-J-1),A2V)
      B1 = U(2,NNO-J)*R(2,NNO-J)
      B2 = U(2,NNO-J-1)*R(2,NNO-J-1)
      CALL A1SUB(Y1,Y2,B0,B1,B2,A1C)
      CALL A2SUB(Y1,Y2,B0,B1,B2,A2C)
C
C     CALCULATE RBUB
   54 UO = D(U(2,NNO),A1U,A2U,YD)
      VO = D(V(2,NNO),A1V,A2V,YD)
      RO = ((C - UO**2 - VO**2)/(C - 1.))**2.5
      CT = 1./SQRT(1. + DYB**2)
      ST = DYB*CT
      RAB = ABS(1./(CT**3*DDYB))
      DB = (DE + XB)/ST
      VS = UO*CT + VO*ST
      VN =-UO*ST + VO*CT
      CALL ARFL(XA+HS*CT,XBT,  YBT, DYBT, DDYBT,2)
      DBT = DB + (1. + DB/(RAB+DB))*VN/VS*HS
      H = XBT - DBT*DYBT/SQRT(1. + DYBT**2) - X
      CL = BO*YD + A1C*YD**2/2. + A2C*YD**3/3. + CLA - CLB
      RBUB = 6.*CL/DB - VS*RO - 4.*VS*RO*CYD
C
      IRBUB=0
      IF(RBUB) 55,55,58
   55 IRBUB = 1
      RETURN
   58 UBS = 0.1
      RBUBP = RBUB**0.4*(C-1.)
C
C     USING NEWTON RAPHSONG METHOD ITERATE ON UB UNTIL RUP = RBUBP
      UB = 0.1
      DO 60 K=1,50
      RUP = C*UB**0.4-UB**2.4
      IF(ABS(RUP-RBUBP).LT..000001) GO TO 70
      DRUPDU = 0.4*C/UB**0.6-2.4*UB**1.4
      UB = UB +(RBUBP-RUP)/DRUPDU
      IF(UB.LT.0) UBS = UBS+.05
      IF(UB.LT.0.) UB = UBS
      IF(UBS.GT.1.) RETURN
   60 CONTINUE
C
      IRBUB = 2
   65 RETURN
C
   70 RB = ((C-UB*UB)/(C-1.))**2.5
      UBINIT = UB
      PB = RB**1.4
C
C     CALCULATE RMB FOR THIS UB
      RMB =  UB/SQRT(1.4*CK*PB/RB)
      IF (YO/Y(2,NN).GE.0.3) NN = NN - 1
      PO = RO**1.4
C
      RMO = SQRT((UO**2 + VO**2)/(1.4*CK*PO/RO))
C
      CALL SUBCRT1(2)
      RETURN
      END
```

```
      SUBROUTINE SUBCRT1(J)
C
C   THIS SUBROUTINE CALCULATES SUBCRITICAL FLOW FOR THE INITIAL PORTION
C   OF THE AIRFOIL SURFACE
C
      COMMON      C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X       ,XA      ,VN      ,VS      ,H       ,DY
     1   ,YO     ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/CCOM/XB      ,YB      ,DYB     ,DDYB    ,DUB     ,PB
     1   ,UB     ,RMB     ,DB      ,HS      ,CRA
      COMMON/DCOM/CS      ,CZ      ,DV1     ,Q1      ,DQ1     ,RK
     1   ,VOO    ,ISKIP
      COMMON/AINPUT/      DUM(12),CXI    ,RUM(11),NN1(7) ,H1(6)
      COMMON/OUTCOM/
     1    XBO(160) ,RMBO(160),DUBO(160),II          ,II2
      COMMON/COMNN/NN
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      DIMENSION BX(4),BU(4,10),BV(4,10),BY(4,10)
C
C   INITIALIZE INPUT
      DCX = 0.004
      CX = CXI -3.*DCX
      CRA = 1.0
      II2=0
      I=0
      N1 = NN - 1
  100 DO 106 K=1,30
C
C   PERFORM FLOW INTEGRATION STEP
      CALL OUNS(1,J)
      DO 104 N = 3, N1
  104 CALL INAS(1,J,N,1)
      CALL INBO(NN,J)
C
      II = II+1
      XBO(II) = XB
      RMBO(II) = RMB
      XX(II,J) = XB
      PP(II,J) = PB
      DUBO(II) = DUB
C
      IF (RMO .GE. 1.0 .OR. DUB .LE. 0.0) RETURN
C
      IF (XB  .GE. CX )  GO TO 120
C
  106 CONTINUE
C
      RETURN
C
C   SAVE FLOW PROPERTIES AT THIS STATION
  120 I=I+1
      BX(I) = X
      DO 124 N = 2, NN
      BU(I,N) = U(J,N)
      BV(I,N) = V(J,N)
  124 BY(I,N) = Y(J,N)
C
      IF (I - 4) 126, 200, 200
  126 CX = CX + DCX
      GO TO 100
C
C   CALCULATE FLOW PROPERTIES AT XB AS INPUT TO NEXT STEP
  200 UO = UB/SQRT(1.+DYB*DYB)
      VO = UO*DYB
      VSQ = UB*UB
      RO = ((C-VSQ)/(C-1.))**2.5
      PO = RO**1.4
      RMO = SQRT(VSQ*RO/(1.4*CK*PO))
```

```
      DO 220 N = 2, NN
      CALL LGRNGN(BY(1,N),BY(2,N),BY(3,N),BY(4,N),
     1BX(1),BX(2),BX(3),BX(4),XB,Y(J,N))
      CALL LGRNGN(BU(1,N),BU(2,N),BU(3,N),BU(4,N),
     1BX(1),BX(2),BX(3),BX(4),XB,U(J,N))
  220 CALL LGRNGN(BV(1,N),BV(2,N),BV(3,N),BV(4,N),
     1BX(1),BX(2),BX(3),BX(4),XB,V(J,N))
C
      II = II+1
      XX(II,J) = XB
      PP(II,J) = PB
      XBO(II) = XB
      RMBO(II) = RMB
      DUBO(II) = DUB
      NN1(7) = II
C
      RETURN
      END
```

```
      SUBROUTINE SUBCRT2(J)
C
C     THIS SUBROUTINE CALCULATES SUBCRITICAL FLOW FOR THE BULK OF THE
C     INTEGRATION PROCESS
C
      COMMON       C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X        ,XA      ,VN      ,VS      ,H       ,DY
     1    ,YO      ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/DCOM/CS       ,CZ      ,DV1     ,Q1      ,DQ1     ,RK
     1    ,YOO     ,ISKIP
      COMMON/AINPUT/      DUM(24),NN1(7) ,H1(3)  ,HO      ,H2(2)
      COMMON/OUTCOM/
     1     AXA(160) ,ARMO(160),ADU(160) ,II       ,II2
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      COMMON/YUVSAV/    NNINIT, NNSPR,NNDWN
     1      ,YUV(90),YU(10),UU(10),VU(10),YL(10),UL(10),VL(10)
     2      ,YOU,YOL,UOU,UOL,VOU,VOL
      COMMON/COMNN/NN
      D(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
C
C     INITIALIZE INPUT
      NTERM = 3
      CX = 0.5
      ISKIP = 2
      CS = 1.0
      CZ = C
      H = HO
      CALL ARFL (XA, X, YO, DY,DDY,2)
      N=NN
      VSO = U(J,NN)*U(J,NN)+V(J,NN)*V(J,NN)
      R(J,NN) = ((C-VSO)/(C-1.))**2.5
      P(J,NN) = R(J,NN)**1.4
      RM(J,NN) = SQRT(VSQ*R(J,NN)/(1.4*CK*P(J,NN)))
C
  250 M = ABS((1.-X)/H)
      DO 268 K = 1, M
      DO 266 KK=1,NTERM
C
C     PERFORM FLOW INTEGRATION STEP
      CALL OUNS(1,J)
      X = X+H
      DO 265 N=3,NN
  265 CALL INAS(1,J,N,NN)
C
      IF(X.GE.CX.OR.X.LT.0.0.OR.RMO.LE.0.1.OR.RMO.GE.1.0.OR.
     1    ABS(ADU(II)).GT.2.0) GO TO 267
C
  266 CONTINUE
C
  267 II = II+1
      AXA(II) = XA
      ARMO(II) = RMO
      ADU(II) = DU(J,NN)
      XX(II,J) = X
      PP(II,J) = PO
C
      IF (X .GE. CX ) GO TO 270
C
      IF (X .LT. 0. .OR. RMO .LE. 0.4.OR.RMO.GE.1.0) RETURN
C
      IF(ABS(ADU(II)).GT.2.0) NN = NN-1
  268 CONTINUE
  270 CONTINUE
      IF(CX.GT.0.999) GO TO 380
C
C     ADD ANOTHER STRIP TO THE INTEGRATION
      Y1 = Y(J,NN-1)-Y(J,NN)
      Y2 = Y(J,NN-2)-Y(J,NN)
```

```
      CALL A1SUB(Y1,Y2,U(J,NN),U(J,NN-1),U(J,NN-2),A1U)
      CALL A2SUB(Y1,Y2,U(J,NN),U(J,NN-1),U(J,NN-2),A2U)
      CALL A1SUB(Y1,Y2,V(J,NN),V(J,NN-1),V(J,NN-2),A1V)
      CALL A2SUB(Y1,Y2,V(J,NN),V(J,NN-1),V(J,NN-2),A2V)
      NN = NN+1
      Y(J,NN) = Y(J,NN-1)/2.
      U(J,NN) = D(U(J,NN-1),A1U,A2U,Y(J,NN)-Y(J,NN-1))
      V(J,NN) = D(V(J,NN-1),A1V,A2V,Y(J,NN)-Y(J,NN-1))
      VSQ = U(J,NN)*U(J,NN)+V(J,NN)*V(J,NN)
      R(J,NN) = ((C-VSQ)/(C-1.))**2.5
      P(J,NN) = R(J,NN)**1.4
      RM(J,NN) = SQRT(VSQ*R(J,NN)/(1.4*CK*P(J,NN)))
      CX = 1.0
      GO TO 250
C
C   SAVE FINAL FLOW PROPERTIES OF THIS SUBROUTINE AS INPUT TO NEXT STEP
  380 II2=1
      NP(J) = II
      IF(J.EQ.2) GO TO 410
      DO 400 N=1,NN
      YU(N) = Y(J,N)
      UU(N) = U(J,N)
  400 VU(N) = V(J,N)
      YOU = YO
      UOU = UO
      VOU = VO
      RETURN
  410 DO 420 N=1,NN
      YL(N) = Y(J,N)
      UL(N) = U(J,N)
  420 VL(N) = V(J,N)
      YOL = YO
      UOL = UO
      VOL = VO
      RETURN
      END
```

```
      SUBROUTINE IOUPRCT
C
C    THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C    OUPUT FOR SUBROUTINE UPRCRIT
C
      COMMON/AINPUT/       AIN(24),NN1(7) ,HI(6)
      COMMON/YUVSAV/    NNINIT,NNSPR,NNDWN,YUV(156)
      COMMON/OUTCOM/
     1     AXB(160) ,ARMB(160),DUM(160) ,II        ,II2
      DIMENSION ISTAR(5),ITITLE(4)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,4)/10HTEST OF CR,10HITICALITY*,10H   UPPER S,
     1   10HURFACE     /
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5),
     1  (ITITLE(I),I=3,4)
      WRITE(6,220) NNINIT,AIN(20)
      WRITE(6,230)  AIN(22),AIN(21)
      CALL UPRCRIT
      IF(II.EQ.0) RETURN
      IHALF = II/2
      J = MOD(II,2)
      IHALF1 = IHALF
      IF(J.EQ.1) IHALF1=IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 25
      DO 20 I=1,IHALF
   20 WRITE(6,300) AXB(I),ARMB(I),AXB(I+IHALF1),ARMB(I+IHALF1)
      IF(J.NE.1) GO TO 28
   25 WRITE(6,300) AXB(IHALF1),ARMB(IHALF1)
   28 RETURN
  200 FORMAT(1H1,4(/),7X,12A10/57X,2A10/)
  220 FORMAT(/20X,34HFROM THE UPSTREAM SOLUTION,    NN =,I2,8H,    DE =,
     1    F10.6)
  230 FORMAT(/20X,36HFROM THE STAGNATION SOLUTION,    YS =,F10.6,
     1 9H,    YSO = ,F10.6)
  300 FORMAT(30X,2(10X,2F10.4))
  400 FORMAT(//29X,2(18X,2HXB,8X,2HMB))
      END
```

```
      SUBROUTINE UPRCRIT
C
C   THIS SUBROUTINE CALCULATES MACH NUMBER FOR A SELECTED NUMBER OF
C   POINTS ON THE INITIAL PORTION OF THE UPPER SURFACE
C
      COMMON       C        ,CK       ,RS       ,FM       ,ALPHA
      COMMON/ACOM/X         ,XA       ,VN       ,VS       ,H        ,DY
     1    ,YO       ,PO       ,RO       ,UO       ,VO       ,RMO      ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/YUVSAV/   NNO,NNSPR,NNDWN
     1        ,YI(10,2),UI(10,2),VI(10,2),YUV(96)
      COMMON/AINPUT/DUM(19),DE    ,YSO    ,YS      ,DDUM(2),NDUM(7)
     1    ,HDUM(6)
      COMMON/OUTCOM/
     1     AXB(160) ,ARMB(150),ADUM(160),II        ,II2
      D(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
C
C   INITIALIZE INPUT
      ND = NNO - 2
      DO 50 N = ND, NNO
      Y(1,N) =YI(N,1)-YSO-YS
      U(1,N) = UI(N,1)
      V(1,N) = VI(N,1)
   50 R(1,N) = ((C-U(1,N)*U(1,N)-V(1,N)*V(1,N))/(C-1.))**2.5
C
      Y1 = Y(1,NNO-1)+YS+YSO-YI(NNO,1)
      Y2 = Y(1,NNO-2)+YS+YSO-YI(NNO,1)
      CALL A1SUB(Y1,Y2,U(1,NNO),U(1,NNO-1),U(1,NNO-2),A1U)
      CALL A2SUB(Y1,Y2,U(1,NNO),U(1,NNO-1),U(1,NNO-2),A2U)
      CALL A1SUB(Y1,Y2,V(1,NNO),V(1,NNO-1),V(1,NNO-2),A1V)
      CALL A2SUB(Y1,Y2,V(1,NNO),V(1,NNO-1),V(1,NNO-2),A2V)
      B0 = U(1,NNO)*R(1,NNO)
      B1 = U(1,NNO-1)*R(1,NNO-1)
      B2 = U(1,NNO-2)*R(1,NNO-2)
      CALL A1SUB(Y1,Y2,B0,B1,B2,A1C)
      CALL A2SUB(Y1,Y2,B0,B1,B2,A2C)
C
      DX = .003
C
   56 XA = 0.
      II = 0
C
      DO 100 I=1,20
      XA = XA+DX
C
C   DETERMINE RBUB FOR THIS XA
      CALL ARFL (XA, XB, YB, DYB, DDYB,1)
      YO = (DE + XB)/DYB + YB
      YD = YO+YSO+YS-YI(NNO,1)
      UO = D(U(1,NNO),A1U,A2U,YD)
      VO = D(V(1,NNO),A1V,A2V,YD)
      VSQ = UO*UO+VO*VO
      IF((C-VSQ).GT.0.) GO TO 57
      IF(I.GT.3.OR.DX.LT.0.0006) RETURN
      DX = .0005
      GO TO 56
   57 RO = ((C-VSQ)/(C-1.))**2.5
      CT = 1./SQRT(1. + DYB**2)
      ST = DYB*CT
      DP = (DE + XB)/ST
      VS = UO*CT + VO*ST
      CL = B0*YD + A1C*YD**2/2. + A2C*YD**3/3.
      RBUB = 2.*CL/DP-VS*RO
C
      IRBUB = 0
      IF(RBUB.LT.0.) GO TO 100
   58 UPS = 0.1
      RBUPP = RBUB**0.4*(C-1.)
C
```

```
C   USING NEWTON-RAPHSONG METHOD, ITERATE ON UB UNTIL RBUBP = RUP
      UB=0.1
      DO 60 K=1,50
      RUP = C*UB**0.4-UB**2.4
      IF(ABS(RUP-RBUBP).LT..000001) GO TO 70
      DRUPDU = 0.4*C/UB**0.6-2.4*UB**1.4
      UB = UB+(RBUBP-RUP)/DRUPDU
      IF(UB.LT.0.) UBS=UBS+.05
      IF(UB.LT.0.) UB=UBS
      IF(UBS.GT.1.) GO TO 100
   60 CONTINUE
C
      IF(I.GT.3.OR.DX.LT.0.0006) RETURN
      DX = .0005
      GO TO 56
   70 RB = ((C-UB**2)/(C-1.))**2.5
      PB = RB**1.4
C
C   CALCULATE RMB FOR THIS UB
      RMB =  UB/SQRT(1.4*CK*PB/RB)
C
      II = II+1
      ARMB(II) = RMB
      AXB(II) = XB
C
  100 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE IOUPRIN(ICRIT)
C
C     THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C     OUTPUT FOR SUBROUTINE UPRINIT
C
      COMMON/AINPUT/      AIN(24),NN1(7) ,HI(6)
      COMMON/YUVSAV/   NNINIT,NNSPR,NNDHN,YUV(155)
      COMMON/OUTCOM/
     1    XBO(160) ,RMBO(160),DUBO(160),II        ,II2
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      COMMON/RBUBCM/RBUB      ,UBINIT    ,IRBUB
      DIMENSION ISTAR(5),ITITLE(4)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,4)/10H* INITIAL ,10HSOLUTION *,10H   UPPER S,
     1 10HURFACE    /
      J=1
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5),
     1  (ITITLE(I), I=3,4)
      WRITE(6,210) NN1(3),AIN(3),AIN(4),AIN(14),HI(2)
      NN1(7) = 0
      IF(HI(2).GT.0.0001.OR.NNINIT.GE.NN1(3)) GO TO 4
      IF(HI(2).LT.0.0001) WRITE(6,360)
      IF(NNINIT.LT.NN1(3)) WRITE(6,370)
      RETURN
    4 CALL UPRINIT(ICRIT)
      IF(IRBUB.EQ.0) WRITE(6,330) RBUB,UBINIT
      IF(IRBUB.EQ.1) WRITE(6,350) RBUB
      IF(IRBUB.EQ.2) WRITE(6,340) RBUB
      IF(II.EQ.0) RETURN
      IF(II2.EQ.0) NN1(7) = II+1
      IF(II2.EQ.0) II=II+1
      IICX = NN1(7)
      IHALF = II/2
      K = MOD(II,2)
      IHALF1 = IHALF
      IF(K.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 15
      DO 10 I=1,IHALF
      IF(I.NE.IICX.AND.(I+IHALF1).NE.IICX) GO TO 5
      IF(I.EQ.IICX) WRITE(6,310) (ISTAR(L),L=1,4),XBO(I+IHALF1),
     1RMBO(I+IHALF1),PP(I+IHALF1,J),DUBO(I+IHALF1)
      IF((I+IHALF1).EQ.IICX) WRITE(6,320) XBO(I),RMBO(I),PP(I,J),DUBO(I)
     1  ,(ISTAR(L),L=1,4)
      GO TO 10
    5 WRITE(6,300) XBO(I),RMBO(I),PP(I,J),DUBO(I),XBO(I+IHALF1),
     1  RMBO(I+IHALF1),PP(I+IHALF1,J),DUBO(I+IHALF1)
   10 CONTINUE
      IF(K.NE.1) GO TO 18
      IF(IICX.EQ.IHALF1) WRITE(6,310) (ISTAR(L),L=1,4)
      IF(IICX.EQ.IHALF1) GO TO 18
   15 WRITE(6,300) XBO(IHALF1),RMBO(IHALF1),PP(IHALF1,J),DUBO(IHALF1)
   18 IF(II2.LE.0) WRITE(6,260)
      RETURN
  200 FORMAT(4(/),      7X,12A10/57X,2A10)
  210 FORMAT(          //20X,4HNN =I2,9H,     XAO =,F10.6,9H,    CYD =F12.8,
     1 9H,    RMC =,F10.6,8H,    HS =,F10.6)
  260 FORMAT(/20X,39H**********INTEGRATION WAS NOT COMPLETED )
  300 FORMAT(10X,2(10X,4F10.4))
  310 FORMAT(20X,4A10,10X,4F10.4)
  320 FORMAT(20X,4F10.4,10X,4A10)
  330 FORMAT( /47X,6HRBUB =,F10.6,10X,4HUB =,F10.6)
  340 FORMAT(/ 39X,6HRBUB =,F10.6,40H******FLOW CONDITIONS CANNOT BE MAT
     1CHED     )
  350 FORMAT(/ 59X,6HRBUB =,F10.6)
  360 FORMAT(//20X,29H**********STEP SIZE TOO SMALL)
  370 FORMAT(//20X,49H**********INSUFFICIENT NUMBER OF STRIPS AVAILABLE)
  400 FORMAT(//10X,2(17X,2HXB,8X,2HMB,8X,2HPB,7X,4HDUDX))
      END
```

```
      SUBROUTINE UPRINIT(ICRIT)
C
C   THIS SUBROUTINE CALCULATES THE INITIAL CONDITIONS ON THE UPPER
C   SURFACE
C
      COMMON        C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X         ,XA      ,VN      ,VS      ,H       ,DY
     1    ,YO      ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/CCOM/XB        ,YB      ,DYB     ,DDYB    ,DUB     ,PB
     1    ,UB      ,RMB     ,DB      ,HS      ,CRA
      COMMON/AINPUT/      ADUM(2),XAO     ,CYD     ,DUM(15),DE
     1    ,YSO     ,YS      ,DDUM(2),NDUM(2),NNUPR   ,MDUM(4),H1
     2    ,HSI     ,H2(4)
      COMMON/YUVSAV/    NNO,NNSPR,NNOWN
     1        ,YI(10,2),UI(10,2),VI(10,2),YUV(96)
      COMMON/COMNN/NN
      COMMON/OUTCOM/EDUM(480),II       ,II2
      COMMON/RBUBCM/RBUB      ,UBINIT    ,IRBUB
      D(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
C
C   INITIALIZE INPUT
      II2=0
      II = 0
      HS = HSI
      NN = NNUPR
      X = -DE
      XA = XAO
      DO 52  N = 1, NNO
      Y(1,N)  =  YI(N,1) -YSO-YS
      U(1,N)  =  UI(N,1)
      V(1,N)  =  VI(N,1)
      VSQ = U(1,N)*U(1,N)+V(1,N)*V(1,N)
      R(1,N)  =  ((C-VSQ)/(C-1.))**2.5
      P(1,N)  =  R(1,N)**1.4
   52 RM(1,N)  = SQRT(VSQ*R(1,N)/(1.4*CK*P(1,N)))
C
C   CALCULATE RBUB
      Y1 = Y(1,NNO-1)+YS+YSO-YI(NNO,1)
      Y2 = Y(1,NNO-2)+YS+YSO-YI(NNO,1)
      CALL A1SUB(Y1,Y2,U(1,NNO),U(1,NNO-1),U(1,NNO-2),A1U)
      CALL A2SUB(Y1,Y2,U(1,NNO),U(1,NNO-1),U(1,NNO-2),A2U)
      CALL A1SUB(Y1,Y2,V(1,NNO),V(1,NNO-1),V(1,NNO-2),A1V)
      CALL A2SUB(Y1,Y2,V(1,NNO),V(1,NNO-1),V(1,NNO-2),A2V)
      B0 = U(1,NNO)*R(1,NNO)
      B1 = U(1,NNO-1)*R(1,NNO-1)
      B2 = U(1,NNO-2)*R(1,NNO-2)
      CALL A1SUB(Y1,Y2,B0,B1,B2,A1C)
      CALL A2SUB(Y1,Y2,B0,B1,B2,A2C)
      CALL ARFL (XA, XB, YB, DYB, DDYB,1)
      YO = (DE + XB)/DYB + YB
      YD = YO+YSO+YS-YI(NNO,1)
      UO = D(U(1,NNO),A1U,A2U,YD)
      VO = D(V(1,NNO),A1V,A2V,YD)
      RO = ((C - UO**2 - VO**2)/(C - 1.))**2.5
      PO = RO**1.4
      RMO = SQRT((UO**2 + VO**2)/(1.4*CK*PO/RO))
      CT = 1./SQRT(1. + DYB**2)
      ST = DYB*CT
      RAB = ABS(1./(CT**3*DDYB))
      DB = (DE + XB)/ST
      VS = UO*CT + VO*ST
      VN =-UO*ST + VO*CT
      CL = B0*YD + A1C*YD**2/2. + A2C*YD**3/3.
      RBUB = 6.*CL/DB - VS*RO - 4.*VS*RO*CYD
C
      II = 0
      IRBUB=0
      IF(RBUB) 55,55,58
```

```
C
   55 IRBUB = 1
      RETURN
C
   58 UBS = 0.1
      RBUBP = RBUB**0.4*(C-1.)
C
C     USING NEWTON-RAPHSONG METHOD, ITERATE ON UB UNTIL RBUBP = RUP
      UB = 0.1
      DO 60 K=1,50
      RUP = C*UB**0.4-UB**2.4
      IF(ABS(RUP-RBUBP).LT..000001) GO TO 70
      DRUPDU = 0.4*C/UB**0.6-2.4*UB**1.4
      UB = UB +(RBUBP-RUP)/DRUPDU
      IF(UB.LT.0.) UBS = UBS+.05
      IF(UB.LT.0.) UB = UBS
      IF(UBS.GT.1.) RETURN
   60 CONTINUE
C
      IRBUB = 2
   65 RETURN
C
   70 IF(C-UB*UB) 65,65,71
   71 RB = ((C-UB*UB)/(C-1.))**2.5
      UBINIT = UB
      PB = RB**1.4
C
C     CALCULATE RMB FOR THIS UB
      RMB =  UB/SQRT(1.4*CK*PB/RB)
C
      CALL ARFL(XA+HS*CT, XBT, YBT, DYBT, DDYBT,1)
      DBT = DB + (1. + DB/(RAB+DB))*VN/VS*HS
      H = XBT - DBT*DYBT/SQRT(1. + DYBT**2) - X
C
      IF(ICRIT.EQ.1) CALL SPRCRT1(1)
      IF(ICRIT.EQ.2) CALL SUBCRT1(1)
C
      RETURN
      END
```

```
      SUBROUTINE SPRCRT1(J)
C
C     THIS SUBROUTINE CALCULATES THE INITIAL FLOW CONDITIONS ON THE UPPER
C     SURFACE
C
      COMMON      C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X       ,XA      ,VN      ,VS      ,H       ,DY
     1    ,YO      ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/CCOM/XB      ,YB      ,DYB     ,DDYB    ,DUB     ,PB
     1    ,UB      ,RMB     ,DB      ,HS      ,CRA
      COMMON/DCOM/CS,CZ,ADDUM(5),ISKIP
      COMMON/AINPUT/      DUM(13),RMC     ,BDUM(4),XASPR  ,CDUM(5)
     1    ,NN1(7) ,H1(6)
      COMMON/YUVSAV/   NNINIT,NNSPR,NNDWN
     1       ,YUV1(60),YSPR(10),USPR(10),VSPR(10),YUV2(66)
      COMMON/OUTCOM/
     1       XBO(160) ,RMBO(160),DUBO(160),II        ,II2
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      COMMON/COMNN/NN
      DIMENSION BVS(6),BVN(6),BDB(6)
      DIMENSION BX(4),BXA(4),BUB(4),BMB(4), BU(4,10),BV(4,10),BY(4,10)
C
C    INITIALIZE INPUT
      II2 = 0
      CRA = 0.
      CSO = 0.9
      DCSO = .02
      DCS1 = .03
      NTERM = 3
      I = 0
      N1 = NN - 1
  100 DO 110 K = 1, 100
      DO 108 KK=1,NTERM
C
C    PERFORM FLOW INTEGRATION STEP IN THE SUBSONIC REGION
      CALL OUNS(1,J)
      DO 104 N = 3, N1
  104 CALL INAS(1,J,N,1)
      CALL INBO(NN,J)
C
      IF(RMB.GE.CSO.OR.DUB.LE.5.0) GO TO 109
  108 CONTINUE
C
  109 II = II+1
      XBO(II) = XB
      RMBO(II) = RMB
      DUBO(II) = DUB
      XX(II,J) = XB
      PP(II,J) = PB
C
      IF(RMB.GE.1..OR.DUB.LE.5.) RETURN
C
      IF(RMB.GE.CSO) GO TO 120
C
  110 CONTINUE
C
      RETURN
C
C    IF THIS IS THE FIRST TIME THROUGH DECREASE THE STEP SIZE BY HALF
  120 IF(I.GT.0) GO TO 122
      HS = HS/2.
      H = H/2.
C
C    SAVE FLOW PROPERTIES AT THIS STATION FOR FUTURE USE
  122 I=I+1
      BX(I) = X
      BXA(I) = XA
      BUB(I) = UB
```

```
            BMB(I) = RMB
            BDB(I) = DB
            BVS(I) = VS
            BVN(I) = VN
            DO 124 N = 2, NN
            BU(I,N) = U(J,N)
            BV(I,N) = V(J,N)
        124 BY(I,N) = Y(J,N)
      C
            IF (I - 4) 126, 150, 150
        126 CSO = CSO + DCSO
            GO TO 100
      C
        150 H = H*2.
            HS = HS*2.
            HX = HS/SQRT(1. + DYB**2)
      C
      C   FIND THE X STATION FOR WHICH RMB IS GREATER THAN 1.03 USING A
      C   LAGRANGIAN FUNCTION
            DO 160 N = 1, 100
            XA = XA + HX
            CALL LGRNGN(BMB(1),BMB(2),BMB(3),BMB(4),
          1BXA(1),BXA(2),BXA(3),BXA(4),XA,RMB)
            IF (RMB .GE. 1.03) GO TO 200
        160 CONTINUE
            RETURN
      C
      C   CALCULATE FLOW PROPERTIES AT THIS X STATION
        200 II2 = -1
            II = II+1
            NN1(7) = II
            CALL LGRNGN(BDB(1),BDB(2),BDB(3),BDB(4),
          1BXA(1),BXA(2),BXA(3),BXA(4),XA,DB)
            CALL LGRNGN(BVS(1),BVS(2),BVS(3),BVS(4),
          1BXA(1),BXA(2),BXA(3),BXA(4),XA,VS)
            CALL LGRNGN(BVN(1),BVN(2),BVN(3),BVN(4),
          1BXA(1),BXA(2),BXA(3),BXA(4),XA,VN)
            CALL ARFL (XA, XB, YB, DYB, DDYB,1)
            CT = 1./SQRT(1. + DYB**2)
            ST = CT*DYB
            YO = YB + DB*CT
            X = XB - DB*ST
            UO = VS*CT - VN*ST
            VO = VS*ST + VN*CT
            RO = ((C - UO*UO - VO*VO)/(C - 1.))**2.5
            PO = RO**1.4
            RMO = SQRT((UO*UO + VO*VO)*RO/(1.4*CK*PO))
            CALL LGRNGN(BUB(1),BUB(2),BUB(3),BUB(4),
          1BXA(1),BXA(2),BXA(3),BXA(4),XA,UB)
            RB = ((C - UB*UB)/(C - 1.))**2.5
            PB = RB**1.4
            RMB = UB/SQRT(1.4*CK*PB/RB)
            DO 220 N = 2, NN
            CALL LGRNGN(BU(1,N),BU(2,N),BU(3,N),BU(4,N),
          1BX(1),BX(2),BX(3),BX(4),X,U(J,N))
            CALL LGRNGN(BV(1,N),BV(2,N),BV(3,N),BV(4,N),
          1 BX(1),BX(2),BX(3),BX(4),X,V(J,N))
            CALL LGRNGN(BY(1,N),BY(2,N),BY(3,N),BY(4,N),
          1 BX(1),BX(2),BX(3),BX(4),X,Y(J,N))
            VSQ = U(J,N)*U(J,N)+V(J,N)*V(J,N)
            R(J,N) = ((C-VSQ)/(C-1.))**2.5
            P(J,N) = R(J,N)**1.4
        220 RM(J,N) = SQRT(VSQ*R(J,N)/(1.4*CK*P(J,N)))
      C
            CS1 = RMB
            I=0
      C
        250 DO 290 K = 1, 50
            DO 280 KK=1,NTERM
      C
```

```
C   PERFORM FLOW INTEGRATION STEP IN SUPERSONIC REGION
      CALL OUNS(1,J)
      N1 = NN - 1
      DO 275 N = 3, N1
  275 CALL INAS(1,J,N,1)
      CALL INBO(NN,J)
C
      IF(RMB.GE.CS1.OR.RMB.LT.1.0.OR.RM(J,NN).GE.RMC) GO TO 285
C
  280 CONTINUE
C
  285 II = II+1
      XBO(II) = XB
      RMBO(II) = RMB
      DUBO(II) = DUB
      XX(II,J) = XB
      PP(II,J) = PB
C
      IF(RMB.LT.1.0.OR.RMB.GT.2.0) RETURN
C
      IF (RMB .GE. CS1)  GO TO 320
C
      IF(RM(J,NN).GE.RMC) NN=NN-1
  290 CONTINUE
C
C   SAVE FLOW PROPERTIES AT THIS STATION
  320 I = I+1
      BX(I) = X
      DO 324 N = 2, NN
      BU(I,N) = U(J,N)
      BV(I,N) = V(J,N)
  324 BY(I,N) = Y(J,N)
C
      IF (I - 4) 326, 350, 350
C
  326 CS1 = CS1 + DCS1
      GO TO 250
C
C   CALCULATE FLOW PROPERTIES AT XB FOR INPUT TO NEXT STEP
  350 YSPR(1) = Y(J,1)
      USPR(1) = U(J,1)
      VSPR(1) = V(J,1)
      DO 360 N = 2, NN
      CALL LGRNGN(BY(1,N),BY(2,N),BY(3,N),BY(4,N),
     1  BX(1),BX(2),BX(3),BX(4),XB,YSPR(N))
      CALL LGRNGN(BU(1,N),BU(2,N),BU(3,N),BU(4,N),
     1  BX(1),BX(2),BX(3),BX(4),XB,USPR(N))
  360 CALL LGRNGN(BV(1,N),BV(2,N),BV(3,N),BV(4,N),
     1  BX(1),BX(2),BX(3),BX(4),XB,VSPR(N))
      NN = NN+1
      YSPR(NN) = YB
      USPR(NN) = UB/SQRT(1.+DYB*DYB)
      VSPR(NN) = USPR(NN)*DYB
      NNSPR = NN
      XASPR = XA
      II2=1
C
      RETURN
      END
```

```
      SUBROUTINE IOSPCT2(J,L)
C
C    THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C    OUTPUT FOR SUBROUTINE SPRCRT2
C
      COMMON/AINPUT/       AIN(24),NN1(7) ,HI(6)
      COMMON/YUVSAV/   NNINIT,NNSPR,NNOWN,YUV(156)
      COMMON/OUTCOM/
     1     AXA(160) ,ADU(160) ,DDQO(160),II        ,II2
      COMMON/COMSPR/ARMO(160)
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      DIMENSION ISTAR(5),ITITLE(4)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,4)/10H* AIRFOIL ,10HSOLUTION *,10H   UPPER S,
     1 10HURFACE     /
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5),
     1  (ITITLE(I),I=3,4)
      WRITE(6,210) NN1(5),AIN(7),(AIN(I),I=15,17),HI(5)
      IF((NNSPR.EQ.0).OR.(NNSPR.GT.9)) GO TO 20
      WRITE(6,220) NNSPR,AIN(19)
      M = (1.0-AIN(7))/0.01+(AIN(7)-0.1)/0.005+(0.1-AIN(3))/HI(5)
      IF(M.LT.470.OR.NNSPR.GE.NN1(5)) GO TO 4
      IF(NNSPR.LT.NN1(5)) WRITE(6,330)
      IF(M.GT.470) WRITE(6,320)
      IF(NN1(5).LT.NNSPR) WRITE(6,330)
      RETURN
    4 DO 5 I=1,160
    5 DDQO(I) = 0.
      CALL INVELOC(L,J)
      CALL SPRCRT2(J,L)
      IF(II.EQ.0) RETURN
      IHALF = II/2
      K = MOD(II,2)
      IHALF1 = IHALF
      IF(K.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 15
      DO 10 I=1,IHALF
   10 WRITE(6,310)AXA(I),ARMO(I),PP(I,J),ADU(I),DDQO(I),AXA(I+IHALF1),
     1  ARMO(I+IHALF1),PP(I+IHALF1,J),ADU(I+IHALF1),DDQO(I+IHALF1)
      IF(K.NE.1) GO TO 18
   15 WRITE(6,310) AXA(IHALF1),ARMO(IHALF1),PP(IHALF1,J),ADU(IHALF1),
     1    DDQO(IHALF1)
   18 IF(II2.EQ.0) WRITE(6,260)
      RETURN
   20 WRITE(6,250)
      RETURN
  200 FORMAT(4(/),7X,12A10/57X,2A10)
  210 FORMAT(         //20X,4HNC =,I2//20X,12HSHOCK LOC. =F10.6,
     1 10H,    BETA = ,F10.6,10H,    DELS =,F10.6,10H,    CDDQ =,F10.6,
     2 8H,    HO =,F10.6)
  220 FORMAT(/20X,31HFROM INITIAL CONDITIONS,    NN =,I2,13H,    X(INIT) =
     1   ,F10.6)
  250 FORMAT(/20X,45H**********PREVIOUS STEP HAS NOT BEEN COMPUTED)
  260 FORMAT(/20X,39H**********INTEGRATION WAS NOT COMPLETED )
  310 FORMAT(2(10X,5F10.4))
  320 FORMAT(//20X,29H**********STEP SIZE TOO SMALL)
  330 FORMAT(//20X,49H**********INSUFFICIENT NUMBER OF STRIPS AVAILABLE)
  400 FORMAT(//2(17X,1HX,9X,2HMO,8X,2HPO,7X,4HDUDX,6X,3HDDQ,1X))
      END
```

```
      SUBROUTINE SPRCRT2(J,L)
C
C   THIS SUBROUTINE PERFORMS FLOW INTEGRATION FOR THE BULK OF THE
C   AIRFOIL SURFACE
C
      COMMON      C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X       ,XA      ,VN      ,VS      ,H       ,DV
     1   ,YO      ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/DCOM/CS      ,CZ      ,DV1     ,Q1      ,DQ1     ,RK
     1   ,VOO     ,ISKIP
      COMMON/AINPUT/      ADUM(6),SL      ,BDUM(7),BETAD   ,DELS
     1   ,CDDQ    ,RKI     ,XASPR   ,CDUM(3),CSI     ,CZI     ,NN1(4)
     2   ,NNSPR   ,NN2(2)  ,H1(4)   ,HO      ,HOWN
      COMMON/YUVSAV/    NNINIT,NNO,NNOWN
     1      ,YUV(60),YSPR(10),USPR(10),VSPR(10),YU(10),UU(10),VU(10)
     2      ,YL(10),UL(10),VL(10),YOU,YOL,UOU,UOL,VOU,VOL
      COMMON/OUTCOM/
     1     AXA(160) ,ADU(160) ,DDQ0(160),II       ,II2
      COMMON/COMPRS/XX(160,2),PP(160,2),NP(2)
      COMMON/COMSPR/ARMO(160)
      D(AO, A1, A2, Z) = AO + A1*Z + A2*Z*Z
C
C   INITIALIZE INPUT
      NN = NNSPR
      II = 0
      II2=0
      NTERM = 3
      ISKIP = 3
      PS = RS**1.4
      CS = 1.0
      CZ = C
      H = HO
      XA = XASPR
      CALL ARFL(XA,X,YO,DV,DUM,J)
      BETA = BETAD/57.2957795
      YO = YSPR(NNO)
      UO = USPR(NNO)
      VO = VSPR(NNO)
      QO = SQRT(UO*UO+VO*VO)
      RO = ((C - QO**2)/(C - 1.))**2.5
      PO = RO**1.4
      RMO =  QO/SQRT(1.4*CK*PO/RO)
      DO 102 N = 1, NN
      Y(J,N) = YSPR(N)
      U(J,N) = USPR(N)
      V(J,N) = VSPR(N)
      VSQ = U(J,N)*U(J,N)+V(J,N)*V(J,N)
      R(J,N) = ((C-VSQ)/(C-1.))**2.5
      P(J,N) = R(J,N)**1.4
  102 RM(J,N) = SQRT(VSQ*R(J,N)/(1.4*CK*P(J,N)))
C
      M = ABS((.99-X)/H)
      DQ1 = 0.0
      DO 295 K=1,M
      DO 280 KK=1,NTERM
      DQ0 = DQ1
C
C   PERFORM FLOW INTEGRATION STEP IN SUPERSONIC REGION
      CALL OUNS(1,J)
      X = X+H
      DO 270 N = 3, NN
  270 CALL INAS(1,J,N,NN)
C
C
      IF(ABS(DUO).GE.100.0.OR.X.LT.0.0.OR.X.GE.SL.OR.RMO.LT.1.)GOTO 285
C
      IF (X .GE. 0.1) H = 0.005
C
```

```
  280 CONTINUE
C
  285 DDQ = (DQ1 - DQ0)/H
      II = II+1
      AXA(II) = XA
      ADU(II) = DU(J,NN)
      ARMO(II) = RMO
      DDQO(II) = DDQ
      XX(II,J) = X
      PP(II,J) = PO

      IF(DDQ.GE.CDDQ) NN = NN-1
C
      IF(ABS(DUO).GE.100..OR.X.LT.0.0) RETURN
C
      IF (RMO .LT. 1.0) RETURN
C
      IF (X .GE. 0.1) H = 0.005
C
      IF(X.GE.SL) GO TO 300
C
  295 CONTINUE
      RETURN
C
C   APPLY RANKINE HUGONIOT RELATIONS THROUGH SHOCK WAVE
  300 QO = SQRT(UO*UO + VO*VO)
      UO1 = QO*SIN(BETA)
      VO1 = QO*COS(BETA)
      RMO = RMO*SIN(BETA)
      R2R1 = 2.4*RMO**2/(0.4*RMO**2 + 2.)
      P2P1 = 1. + 7./6.*(RMO**2 - 1.)
      UO2 = UO1/R2R1
      VO2 = VO1
      QO = SQRT(UO2**2 + VO2**2)
      UO = QO/SQRT(1. + DY*DY)
      VO = UO*DY
      RO = RO*R2R1
      PO = PO*P2P1
C
      IF (DELS .LE. 0.0) GO TO 306
      PS2 = PS*(2.4*RMO**2/(0.4*RMO**2 + 2.))**3.5/P2P1**2.5
      RS2 = RO*(PS2/PO)**(1./1.4)
      CZ = (C - 1.)*PS2/RS2
      CS = PO/RO**1.4
C
  306 H = 0.01
      RMO = SQRT((UO*UO + VO*VO)/(1.4*CK*PO/RO))
      IF(L.EQ.2) CALL LGRNGN(VO,V(J,NN-1),V(J,NN-2),V(J,NN-3),0.0,
     1   Y(J,NN-1)-YO,Y(J,NN-2)-YO,Y(J,NN-3)-YO,Y(J,NN)-YO,V(J,NN))
C
      M = (1.0-X)/H
      DO 320 K = 1, M
      DO 308 KK=1,NTERM
C
C   PERFORM FLOW INTEGRATION STEP IN SUBSONIC REGION
      CALL OUNS(1,J)
      X = X+H
      DO 307 N = 3, NN
  307 CALL INAS(1,J,N,NN)
C
      IF(RMO.GT.1.0.OR.X.GE.1.0.OR.ABS(DUO).GE.50.0.OR.X.LT.0.)GO TO 310
  308 CONTINUE
C
  310 II = II+1
      AXA(II) = XA
      ADU(II) = DU(J,NN)
      ARMO(II) = RMO
      XX(II,J) = X
      PP(II,J) = PO
C
```

```
      IF (ABS(DUO    ) .GE.50.0 .OR. X .LT. 0.0.OR.RHO.GT.1.0) RETURN
C
      IF(X.GE.1.0) GO TO 360
C
  320 CONTINUE
C
C   SAVE FLOW PROPERTIES CALCULATED AT FINAL STATION AS INPUT TO NEXT
C   STEP
  360 II2=1
      NP(J) = II
      NNDWN = NN
      DO 400 N=1,NN
      YU(N) = Y(J,N)
      UU(N) = U(J,N)
  400 VU(N) = V(J,N)
      YOU = YO
      UOU = UO
      VOU = VO
      CSI = CS
      CZI = CZ
      RETURN
      END
```

```
      SUBROUTINE IODNSTM(J)
C
C   THIS SUBROUTINE PRINTS THE INPUT PARAMETERS AND THE CALCULATED
C   OUTPUT FOR SUBROUTINE DWNSTRM
C
      COMMON/AINPUT/      AIN(24),NN1(7) ,HI(6)
      COMMON/OUTCOM/
     1    AX(160)   ,APO(160) ,AP1(160) ,II        ,II1
      COMMON/COMDWN/ARMO(160)
      COMMON/YUVSAV/NNINIT    ,NNSPR    ,NNDWN    ,YUV(156)
      DIMENSION ISTAR(5),ITITLE(2)
      DATA (ISTAR(I),I=1,5)/5*10H**********/
      DATA (ITITLE(I),I=1,2)/10HDOWNSTREAM,10H SOLUTION*/
      WRITE(6,200) (ISTAR(I),I=1,5),(ITITLE(I),I=1,2),(ISTAR(I),I=1,5)
      WRITE(6,210) NN1(6),HI(6),AIN(18)
      WRITE(6,220) AIN(7),AIN(15),AIN(23),AIN(24)
      M = 9.0/HI(6)
      IF(M.LT.470.OR.NNDWN.GE.NN1(6)) GO TO 4
      IF(M.GT.470) WRITE(6,310)
      IF(NNDWN.LT.NN1(6)) WRITE(6,320)
      RETURN
    4 CALL DWNSTRM(J)
      IF(II.EQ.0) RETURN
      IHALF = II/2
      K = MOD(II,2)
      IHALF1 = IHALF
      IF(K.EQ.1) IHALF1 = IHALF+1
      WRITE(6,400)
      IF(II.EQ.1) GO TO 15
      DO 10 I=1,IHALF
      WRITE(6,300) AX(I),ARMO(I),APO(I),AP1(I),AX(I+IHALF1),ARMO(I+IHALF
     1  1),APO(I+IHALF1),AP1(I+IHALF1)
   10 CONTINUE
      IF(K.NE.1) GO TO 18
   15 WRITE(6,300) AX(IHALF1),ARMO(IHALF1),APO(IHALF1),AP1(IHALF1)
   18 CONTINUE
      IF(II1.EQ.0) WRITE(6,250)
      RETURN
  200 FORMAT(1H1,4(/),7X,12A10)
  210 FORMAT(          //20X,4HNN =,I2,7H,    H =,F10.6,8H,    RK =F10.6)
  220 FORMAT(/20X,46HFROM UPPER SURFACE INTEGRATION,    SHOCK LOC. =
     1  F8.4 ,10H,    BETA = F8.4 ,3H,    CS = F8.4 ,8H,    CZ = F8.4 )
  250 FORMAT(/20X,39H**********INTEGRATION HAS NOT COMPLETED )
  300 FORMAT(10X,2(10X,4F10.4))
  310 FORMAT(//20X,29H**********STEP SIZE TOO SMALL)
  320 FORMAT(//20X,49H**********INSUFFICIENT NUMBER OF STRIPS AVAILABLE)
  400 FORMAT(//8X,2(19X,1HX,9X,2HMO,8X,2HPO,8X,2HP1))
      END
```

```
      SUBROUTINE DWNSTRM(J)
      COMMON       C         ,CK       ,RS       ,FM       ,ALPHA
      COMMON/ACOM/X          ,XA       ,VN       ,VS       ,H        ,DY
     1   ,YO      ,PO        ,RO       ,UO       ,VO       ,RMO      ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/DCOM/CS          ,CZ       ,DV1      ,Q1       ,DQ1      ,RK
     1   ,VOO     ,ISKIP
      COMMON/AINPUT/      AIN(17),RKI      ,DDUM(4),CSI      ,CZI
     1   ,NN1(5) ,NN        ,NN2      ,HT(5)    ,HI
      COMMON/YUVSAV/    NNINIT,NNSPR,NNDWN
     1      ,YUV(60),YSPR(10),USPR(10),VSPR(10),YU(10),UU(10),VU(10)
     2        ,YL(10),UL(10),VL(10),YOU,YOL,UOU,UOL,VOU,VOL
      COMMON/OUTCOM/
     1     AX(160)    ,APO(160)  ,AP1(160) ,II        ,II1
      COMMON/COMDWN/ARMO(160)
C
C   INITIALIZE INPUT
      NTERM = 4
      CS = CSI
      CZ = CZI
      H = HI
      RK = RKI
      II = 0
      II1=0
      ISKIP = 1
      IF(J.EQ.1) GO TO 4
      DO 3 N=1,NN
      YU(N) = YL(N)
      UU(N) = UL(N)
    3 VU(N) = VL(N)
    4 DO 5 N=1,NN
      Y(1,N) = YU(N)
      U(1,N) = UU(N)
    5 V(1,N) = VU(N)
      X = 1.
      DO 14 N = 1, NN
      VSQ = UU(N)*UU(N)+VU(N)*VU(N)
      R(1,N) = ((C-VSQ)/(C-1.))**2.5
      P(1,N) = R(1,N)**1.4
      RM(1,N) = SQRT(VSQ/(1.4*CK*P(1,N)/R(1,N)))
   14 DU(1,N) = 0.
      VSQ = UOU*UOU+VOU*VOU
      IF(J.EQ.2) VSQ = UOL*UOL+VOL*VOL
      RO   = ((CZ - VSQ)/(CS*(C-1.)))**2.5
      PO   = RO **1.4*CS
      RMO  =SQRT(VSQ/(1.4*CK*PO/RO))
      DUO  = 0.
      YO = (YOU-YOL)/2.
      QOU = SQRT(VSQ)
      TU = ATAN(VOU/UOU)
C
C   DETERMINE T AND VOO- INITIAL VALUES
      TL =-ATAN(VOL/UOL)
      T = (TU + TL)/2.
      UO   = QOU*COS(T)
      VO   = QOU*SIN(T)
      VOO = VO
      M = ABS(10./H/NTERM)
      DO 40 K = 1, M
      DO 35 KK=1,NTERM
C
C   PERFORM FLOW INTEGRATION STEP
      CALL OUNS(1,1)
      CALL INAS(1,1,NN,NN)
C
      IF(RM(1,NN).GT.0.95) GO TO 36
      IF (X .GE.10..OR. X .LT .0.0 .OR. RMO  .GE. 0.97) GO TO 36
      X = X + H
   35 CONTINUE
```

```
36 II = II+1
   AX(II) = X
   ARMO(II) = RMO
   APO(II) = PO
   AP1(II) = P(1,NN)
   IF(X.GE.10.) GO TO 50
   IF(RM(1,NN).GT.0.95.OR.X.LT.0.0.OR.RMO.GE.0.97) RETURN
40 CONTINUE
50 II1=1
   RETURN
   END
```

```
      SUBROUTINE AKUTTA
C
C   THIS SUBROUTINE PRINTS THE CALCULATED PRESSURE DISTRIBUTION ON THE
C   UPPER AND LOWER SURFACES
C
      COMMON/COMPRS/XX(160,2),PP(160,2),NN1      ,NN2
      DIMENSION ISTAR(4),ITITLE(4)
      DATA (ISTAR(I),I=1,4)/4*10H**********/
      DATA (ITITLE(I),I=1,4)/10H**** PARTI,10HAL PRESSUR,10HE DISTRIBU,
     110HTION *****/
      WRITE(6,200) (ISTAR(I),I=1,4),(ITITLE(I),I=1,4),(ISTAR(I),I=1,4)
      IF(NN1.EQ.0.OR.NN2.EQ.0) RETURN
      WRITE(6,400)
      N1 = NN1
      N2 = NN2
      N1HALF = N1/2
      N2HALF = N2/2
      J1 = MOD(N1,2)
      J2 = MOD(N2,2)
      IHALF = N1HALF
      IF(N1.GT.N2) IHALF = N2HALF
      IF(J1.EQ.1) N1HALF = N1HALF+1
      IF(J2.EQ.1) N2HALF = N2HALF+1
      DO 10 I=1,IHALF
   10 WRITE(6,300) XX(I,1),PP(I,1),XX(I+N1HALF,1),PP(I+N1HALF,1),
     1             XX(I,2),PP(I,2),XX(I+N2HALF,2),PP(I+N2HALF,2)
      IF(N1.GT.N2) GO TO 30
      IF(J1.NE.1) GO TO 18
      WRITE(6,310)  XX(N1HALF,1),PP(N1HALF,1),XX(N1HALF,2),PP(N1HALF,2),
     1             XX(N1HALF+N2HALF,2),PP(N1HALF+N2HALF,2)
   18 N2STOP = N2HALF
      IF(J2.EQ.1) N2STOP = N2HALF-1
      NSTART = N1HALF+1
      DO 19 I=NSTART,N2STOP
   19 WRITE(6,320) XX(I,2),PP(I,2),XX(I+N2HALF,2),PP(I+N2HALF,2)
      IF(J2.NE.1) GO TO 50
      WRITE(6,320) XX(N2HALF,2),PP(N2HALF,2)
      GO TO 50
   30 WRITE(6,340) XX(N2HALF,1),PP(N2HALF,1),XX(N1HALF+N2HALF,1),
     1             PP(N1HALF+N2HALF,1),XX(N2HALF,2),PP(N2HALF,2)
   38 N1STOP = N1HALF
      IF(J1.EQ.1) N1STOP = N1HALF-1
      NSTART = N2HALF+1
      DO 39 I=NSTART,N1STOP
   39 WRITE(6,340) XX(I,1),PP(I,1),XX(I+N1HALF,1),PP(I+N1HALF,1)
      IF(J1.NE.1) GO TO 50
      WRITE(6,340) XX(N1HALF,1),PP(N1HALF,1)
   50 CONTINUE
      RETURN
  200 FORMAT(1H1,4(/)7X,12A10)
  300 FORMAT(12X,2(10X,4F10.6))
  310 FORMAT(22X,2F10.6,30X,4F10.6)
  320 FORMAT(72X,4F10.6)
  340 FORMAT(22X,4F10.6,10X,2F10.6)
  400 FORMAT(/35X,13HUPPER SURFACE,37X,13HLOWER SURFACE//10X,
     1   2(10X,2(8X,1HX,9X,2HP0)))
      END
```

```
      SUBROUTINE INVELOC(L,J)
      COMMON/COMNN/NN
      COMMON/ACOM/X,XA,WN,VS,H,DY,YO,PO,RO,UO,VO,RMO,DUO,
     1 Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/YUVSAV/       NNINIT,NNO,NNOWN
     1     ,YUV1(60),YSPR(10),USPR(10),VSPR(10),YUV2(66)
      D(A0,A1,A2,Z) = A0+A1*Z+A2*Z*Z
      IF(J.EQ.2) GO TO 2
      NN = NNO-1
      DO 1 K=1,NN
      Y(J,K) = YSPR(K)
      U(J,K) = USPR(K)
    1 V(J,K) = VSPR(K)
      YO = YSPR(NNO)
      UO = USPR(NNO)
      VO = VSPR(NNO)
    2 IF(L.EQ.1) GO TO 4
      IF(L.GE.3) GO TO 3
      WRITE(6,310)
      CALL LGRNGN(VO,V(J,NN-1),V(J,NN-2),V(J,NN-3),0.0,Y(J,NN-1)-YO,
     1   Y(J,NN-2)-YO,Y(J,NN-3)-YO,Y(J,NN)-YO,V(J,NN))
      GO TO 5
    3 WRITE(6,320)
      Y1 = Y(J,NN-2)-Y(J,NN-1)
      Y2 = Y(J,NN-3)-Y(J,NN-1)
      CALL A1SUB(Y1,Y2,V(J,NN-1),V(J,NN-2),V(J,NN-3),A1V)
      CALL A2SUB(Y1,Y2,V(J,NN-1),V(J,NN-2),V(J,NN-3),A2V)
      V(J,NN) = D(V(J,NN-1),A1V,A2V,Y(J,NN)-Y(J,NN-1))
      GO TO 5
    4 WRITE(6,300)
    5 WRITE(6,400)
      NNP1 = NN+1
      IHALF = NNP1/2
      K = MOD(NNP1,2)
      IHALF1 = IHALF
      IF(K.EQ.1) IHALF1=IHALF+1
      DO 10 I=1,IHALF
      IF((I+IHALF1).EQ.NNP1) GO TO 7
    6 WRITE(6,410) Y(J,I),U(J,I),V(J,I),Y(J,I+IHALF1),U(J,I+IHALF1),
     1   V(J,I+IHALF1)
      GO TO 10
    7 WRITE(6,410) Y(J,I),U(J,I),V(J,I),YO,UO,VO
   10 CONTINUE
      IF(K.EQ.0) GO TO 12
      WRITE(6,410) Y(IHALF1,J),U(IHALF1,J),V(IHALF1,J)
   12 IF(J.EQ.1) VSPR(NN) = V(J,NN)
      RETURN
  300 FORMAT(//47X,40H***INTERMEDIATE VELOCITY DISTRIBUTION***)
  310 FORMAT(//34X,66H***INTERMEDIATE VELOCITY DISTRIBUTION USING LAGRAN
     1GIAN FUNCTION***)
  320 FORMAT(//34X,65H***INTERMEDIATE VELOCITY DISTRIBUTION USING PARABO
     1LIC FUNCTION***)
  400 FORMAT(/17X,2(19X,1HY,9X,1HU,9X,1HV))
  410 FORMAT(20X,2(10X,3F10.6))
      END
```

```
      SUBROUTINE ARFL(XA,XB,YB,DYB,DDYB,J)
C
C    THIS SUBROUTINE DETERMINES THE Y COORDINATE AND ITS FIRST AND
C    SECOND DERIVATIVES AT A POINT ON THE AIRFOIL
C
      COMMON/PTARFL/XX(40,2) ,YY(40,2) ,AM(40,2) ,CA       ,SA
      IF(XA.GE.1.0) GO TO 60
      DO 10 I=1,40
      IF(XA-XX(I,J))20,20,10
   10 CONTINUE
      XA = 10000.
      RETURN
   20 IF(I.GT.1) GO TO 30
      XA = 0.001
      YA=0.001
      DYA = AM(1,J)
      DDYA = (-4.*AM(1,J)-2.*AM(2,J)+6.*(YY(2,J)-YY(1,J))/
     1   (XX(2,J)-XX(1,J)))/(XX(2,J)-XX(1,J))
      GO TO 40
   30 H = XX(I,J)-XX(I-1,J)
      X2MXA = XX(I,J)-XA
      XAMX1 = XA-XX(I-1,J)
      YA = AM(I-1,J)*X2MXA**2*XAMX1/H**2
      YA = YA-AM(I,J)*XAMX1**2*X2MXA/H**2
      YA = YA+YY(I-1,J)*X2MXA**2*(2.*XAMX1+H)/H**3
      YA = YA+YY(I,J)*XAMX1**2*(2.*X2MXA+H)/H**3
      DYA = AM(I-1,J)*X2MXA*(X2MXA-2.*XAMX1)/H**2
      DYA = DYA-AM(I,J)*XAMX1*(2.*X2MXA-XAMX1)/H**2
      DYA = DYA+6.*(YY(I,J)-YY(I-1,J))*X2MXA*XAMX1/H**3
      DDYA = -2.*AM(I-1,J)*(2.*X2MXA-XAMX1)/H**2
      DDYA = DDYA+2.*AM(I,J)*(2.*XAMX1-X2MXA)/H**2
      DDYA = DDYA+6.*(YY(I,J)-YY(I-1,J))*(X2MXA-XAMX1)/H**3
   40 IF(J.EQ.2) GO TO 50
      XB = XA*CA+YA*SA
      YB = YA*CA-XA*SA
      DYB = (DYA*CA-SA)/(CA+DYA*SA)
      DDYB = DDYA*(CA-DYB*SA)**3
      RETURN
   50 XB = XA*CA-YA*SA
      YB = YA*CA+XA*SA
      DYB = (DYA*CA+SA)/(CA-DYA*SA)
      DDYB = DDYA/(CA-SA*DYA)**3
      RETURN
   60 XB = 1.0
      YB = SA/CA
      DYB = YB
      DDYB = 0.
      RETURN
      END
```

```
      SUBROUTINE LGRNGN(A1,A2,A3,A4,X1,X2,X3,X4,X,ANS)
      F1 = X-X1
      F2 = X-X2
      F3 = X-X3
      F4 = X-X4
      F12 = X1-X2
      F13 = X1-X3
      F14 = X1-X4
      F21 = X2-X1
      F23 = X2-X3
      F24 = X2-X4
      F31 = X3-X1
      F32 = X3-X2
      F34 = X3-X4
      F41 = X4-X1
      F42 = X4-X2
      F43 = X4-X3
      D1 = F12*F13*F14
      D2 = F21*F23*F24
      D3 = F31*F32*F34
      D4 = F41*F42*F43
      U1 = F2*F3*F4
      U2 = F1*F3*F4
      U3 = F1*F2*F4
      U4 = F1*F2*F3
      ANS1 = A1*U1/D1+A2*U2/D2
      ANS2 = A3*U3/D3+A4*U4/D4
      ANS = ANS1+ANS2
      RETURN
      END
```

```
      SUBROUTINE A1SUB(Y1,Y2,U0,U1,U2,ANS)
      F1 = Y2*Y2*U1
      F2 = Y1*Y1*U2
      F3 = Y2*Y2-Y1*Y1
      F4 = Y2-Y1
      F5 = Y1*Y2
      ANS1 = F1-F2-F3*U0
      ANS2 = F4*F5
      ANS = ANS1/ANS2
      RETURN
      END
```

```
      SUBROUTINE A2SUB(Y1,Y2,U0,U1,U2,ANS)
      F1 = -Y2*U1
      F2 = Y1*U2
      F3 = Y2-Y1
      F4 = Y1*Y2
      ANS1 = F1+F2+F3*U0
      ANS2 = F3*F4
      ANS = ANS1/ANS2
      RETURN
      END
```

```
      SUBROUTINE DIST(M,I,N,DY1,DVS,DV1)
C
C    THIS SUBROUTINE PERFORMS A FLOW INTEGRATION STEP ON THE STAGNATION
C    STREAMLINE IN THE UPSTREAM SOLUTION
C    THIS SUBROUTINE INCLUDES THE EFFECTS OF THE CROSS VELOCITY GRADIENT
C    ,DVOO, IN DETERMINING THE FLOW CONDITIONS FAR UPSTREAM FROM THE AIRFOIL
C
      COMMON        C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X         ,XA      ,VN      ,VS      ,H       ,DY
     1   ,YO     ,PO        ,RO      ,UO      ,VO      ,RMO     ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/BCOM/          XO      ,DVOO    ,L
      COMMON/DCOM/          DRHU(2),DPRU(2),DRUV(2)
      DIMENSION ZVS(5),ZV1(5),ZY1(5)
      DATA ZV1,ZVS,ZY1/ 15*0.0   /
      J=2
      IF(I.EQ.2) J=1
      DO 700 K=2,5
      RN = (K-1)/2
      RNH = RN/2.*H
      AY1 = YO*M+ZY1(K-1)*RNH
      AVS = VS+ZVS(K-1)*RNH
      AV1 = VO*M
      IF (AVS - AV1) 200, 100, 100
  100 IF     (C - AVS**2) 200, 300, 300
  200 X = -1.
      RETURN
  300 AR1 = ((C - AVS**2            )/(C - 1.))**2.5
      AP1 = AR1**1.4
      AU1 = SQRT(AVS**2 - AV1**2)
      AX = X + RNH
      Y10 = AY1+Y(I,N)
      Y20 = Y(J,N)+Y(I,N)
      Y21 = Y(J,N)-AY1
      Y10SQ = Y10*Y10
      Y20SQ = Y20*Y20
      Y10QU = Y10SQ*Y10SQ
      Y20QU = Y20SQ*Y20SQ
      DY   = -V(I,N)/U(I,N)
      DY1 = AV1/AU1
      DY2 = V(J,N)/U(J,N)
      D = Y10*Y20*Y21
      DD = -Y21*(Y20+Y10)*DY + Y20*(Y21-Y10)*DY1    +Y10*(Y21+Y20)*DY2
      EY2 = Y20SQ-Y10SQ
      ALOC = R(I,N)*U(I,N)
      ALOY = -ALOC*V(I,N)
      AL1C = AR1*AU1 - ALOC
      AL1Y = AR1*AU1*AV1 - ALOY
      AL2C = R(J,N)*U(J,N) -ALOC
      AL2Y = R(J,N)*U(J,N)*V(J,N) -ALOY
      A1C = (AL1C*Y20SQ-AL2C*Y10SQ)/D
      A1Y = (AL1Y*Y20SQ-AL2Y*Y10SQ)/D
      A2C = (AL2C*Y10 - AL1C*Y20)/D
      A2Y = (AL2Y*Y10 - AL1Y*Y20)/D
      F1  = Y10*(D-EY2*Y10/2.+Y10SQ*Y21/3.)
      F2 = Y10SQ*Y20*(Y20/2.-Y10/3.)
      FC1 = AL1C+2.*AL2C
      FC2 = Y20 -Y10/3.
      FC3 = AL1Y+2.*AL2Y
      FC4 = Y10/2.-Y20/3.
      FC5 = AL1C-AL2C
      FC6 = Y10-Y20/3.
      FC7 = A1C/2.+Y20*A2C/3.
      FC8 = AL1Y-AL2Y
      FC9 = A1Y/2.+Y20*A2Y/3.
      F3C = -Y10QU*DRHU(J)/6.
      F3C = F3C +(-AL1C*D+Y10SQ*(Y10*FC1/3.-Y20*AL1C))*DY
      F3C = F3C -2.*AL2C*Y10SQ*Y10/3.*DY1
      F3C = F3C +AL1C*Y10SQ*FC2*DY2
      F3C = F3C -Y10SQ*(A1C/2.+Y10*A2C/3.)*DD
```

```
      F3C = F3C +(AR1*AV1+R(I,N)*V(I,N))*D
      F3Y = -Y10QU*DRUV(J)/6.
      F3Y = F3Y +(-AL1Y*D +Y10SQ*(Y10*FC3/3.-Y20*AL1Y))*DY
      F3Y = F3Y -2.*AL2Y*Y10SQ*Y10/3.*DY1
      F3Y = F3Y +AL1Y*Y10SQ*FC2*DY2
      F3Y = F3Y -Y10SQ*(A1Y/2.+Y10*A2Y/3.)*DD
      F3Y = F3Y +(CK*AP1+AR1*AV1*AV1-CK*P(I,N)-R(I,N)*V(I,N)*V(I,N))*D
      F4 = Y20*(D + Y20*(-EY2/2. + Y20*Y21/3.))
      F5  = Y20QU/6.
      F6C = -Y10*Y20SQ*FC4*DRHU(J)
      F6C = F6C +(-AL2C*D+Y20SQ*(AL2C*Y10-AL1C*Y20+Y20*FC5/3.))*DY
      F6C = F6C -Y20SQ*(AL2C*FC6*DY1-2.*Y20*AL1C/3.*DY2+FC7*DD)
      F6C = F6C +(R(J,N)*V(J,N)+R(I,N)*V(I,N))*D
      F6Y = -Y10*Y20SQ*FC4*DRUV(J)
      F6Y = F6Y +(-AL2Y*D +Y20SQ*(AL2Y*Y10-AL1Y*Y20+Y20*FC8/3.))*DY
      F6Y = F6Y -Y20SQ*(AL2Y*FC6*DY1 -2.*Y20*AL1Y/3.*DY2+FC9*DD)
      F6Y = F6Y +(CK*(P(J,N)-P(I,N)) +R(J,N)*V(J,N)*V(J,N) -R(I,N)*
     1     V(I,N)*V(I,N))*D
      DEL = F1*F5 - F2*F4
      E2 = F4*F3C - F1*F6C
      E6 = F4*F3Y - F1*F6Y
      ZVS(K) = DV00*(AX/X0)**L/(AVS**2*AR1/(1.4*CK*AP1)-1.)
      ZV1(K) = (E6 -AV1*E2)/(AR1*AU1*DEL)
  700 ZY1(K) = DY1
      DVS    = (ZVS(2) + 2.*(ZVS(3) + ZVS(4)) + ZVS(5))/6.
      DV1    = (ZV1(2) + 2.*(ZV1(3) + ZV1(4)) + ZV1(5))/6.
      DY1    = (ZY1(2) + 2.*(ZY1(3) + ZY1(4)) + ZY1(5))/6.
      RETURN
      END
```

```
      SUBROUTINE STNR(N,T,DY,DVS)
C
C   THIS SUBROUTINE PERFORMS AFLOW INTEGRATION STEP ON THE STAGNATION
C   STREAMLINE IN THE UPSTREAM SOLUTION
C   THIS SUBROUTINE NEGLECTS THE EFFECTS OF CHANGES IN THE VERTICAL
C   COMPONENT OF THE STAGNATION STREAMLINE IN THE FLOW INTEGRATION
C
      COMMON        C        ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X          ,XA      ,VN      ,VS      ,H        ,DY1
     1   ,YO      ,PO       ,RO      ,UO      ,VO      ,RMO      ,DUO
     2.  ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/OCOM/          DRHU(2),DPRU(2),DRUV(2)
      DIMENSION ZY(5),ZVS(5),ZT(5)
      DATA ZY,ZVS,ZT/15*0.0/
      DO 700 K=2,5
      RN = (K-1)/2
      RNH = RN/2.*H
      AVS = VS+ZVS(K-1)*RNH
      AY  = YO+ZY(K-1)*RNH
      AT  = T +ZT(K-1)*RNH
      IF     (C - AVS**2) 100, 300, 300
  100 X = -1.
      RETURN
  300 AR  = ((C - AVS**2            )/(C - 1.))**2.5
      AP  = AR **1.4
      Y10 = AY+Y(2,N)
      Y20 = Y(1,N)+Y(2,N)
      ST = SIN(AT)
      CT = COS(AT)
      DY = ST/CT
      VNO = -U(2,N)*ST -V(2,N)*CT
      VN2 = -U(1,N)*ST +V(1,N)*CT
      VSO = U(2,N)*CT -V(2,N)*ST
      VS2 = U(1,N)*CT +V(1,N)*ST
      CALL A1SUB(Y10,Y20, R(2,N)*VSO*VNO,0.0,R(1,N)*VS2*VN2,A1RUV)
      CALL A2SUB(Y10,Y20, R(2,N)*VSO*VNO,0.0,R(1,N)*VS2*VN2,A2RUV)
      DVDY =(A1RUV + 2.*A2RUV*Y10) / (AR *AVS)
      CALL A1SUB(Y10,Y20,P(2,N),AP,P(1,N),A1P)
      CALL A2SUB(Y10,Y20,P(2,N),AP,P(1,N),A2P)
      DPDY = A1P  + 2.*A2P*Y10
      ZVS(K) = DVDY/ (AVS*AVS*AR/(1.4*CK*AP)-1.)
      ZT(K)  = CK/(AR*AVS**2)*(DY*(-AR*AVS/CK)*ZVS(K)-(1.+DY**2)*DPDY)
  700 ZY(K)  = DY
      DVS    = (ZVS(2) + 2.*(ZVS(3) + ZVS(4)) + ZVS(5))/6.
      DY     = (ZY (2) + 2.*(ZY (3) + ZY (4)) + ZY (5))/6.
      DT     = (ZT (2) + 2.*(ZT (3) + ZT (4)) + ZT (5))/6.
      X = X + H
      VS = VS + H*DVS
      YO = YO+H*DY
      T = T + H*DT
      UO = VS*COS(T)
      VO = VS*SIN(T)
      RO = ((C-VS**2)/(C-1.))**2.5
      PO = RO**1.4
      RMO= VS/SQRT(1.4*CK*PO/RO)
      RETURN
      END
```

```
      SUBROUTINE LUMR(M,I,N,DY1,DVS,DV1)
C
C    THIS SUBROUTINE PERFORMS A FLOW INTEGRATION STEP ON THE STAGNATION
C    STREAMLINE IN THE UPSTREAM SOLUTION
C
      COMMON      C        ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X        ,XA      ,VN      ,VS      ,H        ,DY
     1    ,YO      ,PO      ,RO      ,UO      ,VO      ,RMO      ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/OCOM/         DRHU(2),DPRU(2),DRUV(2)
      DIMENSION ZVS(5),ZV1(5),ZY1(5)
      DATA ZY1,ZVS,ZV1/15*0.0/
      J=2
      IF(I.EQ.2) J=1
      DO 700 K=2,5
      RN = (K-1)/2
      RNH = RN/2.*H
      AY1 = YO*M+ZY1(K-1)*RNH
      AVS = VS +ZVS(K-1)*RNH
      AV1 = VO*M
      IF (AVS - AV1) 200, 100, 100
  100 IF     (C - AVS**2) 200, 300, 300
  200 X = -1.
      RETURN
  300 AR1 = ((C - AVS**2            )/(C - 1.))**2.5
      AP1 = AR1**1.4
      AU1 = SQRT(AVS**2 - AV1**2)
      Y10 = AY1+Y(I,N)
      Y20 = Y(J,N)+Y(I,N)
      Y21 = Y(J,N)-AY1
      Y10SQ = Y10*Y10
      Y20SQ = Y20*Y20
      Y10QU = Y10SQ*Y10SQ
      Y20QU = Y20SQ*Y20SQ
      DY = -V(I,N)/U(I,N)
      DY1 = AV1/AU1
      DY2 = V(J,N)/U(J,N)
      CT = 1.0/SQRT(1. + DY1**2)
      ST = DY1/SQRT(1. + DY1**2)
      VNO = -U(I,N)*ST-V(I,N)*CT
      VN2 = -U(J,N)*ST+V(J,N)*CT
      VSO =  U(I,N)*CT-V(I,N)*ST
      VS2 =  U(J,N)*CT+V(J,N)*ST
      CALL A1SUB(Y10,Y20, R(I,N)*VSO*VNO,0.0,R(J,N)*VS2*VN2,A1RUV)
      CALL A2SUB(Y10,Y20, R(I,N)*VSO*VNO,0.0,R(J,N)*VS2*VN2,A2RUV)
      DVDY =(A1RUV + 2.*A2RUV*Y10) / (AR1*AVS)
      D = Y10*Y20*Y21
      DD = -Y21*(Y20+Y10)*DY + Y20*(Y21-Y10)*DY1    +Y10*(Y21+Y20)*DY2
      EY2 = Y20SQ-Y10SQ
      ALOC = R(I,N)*U(I,N)
      ALOY = -ALOC*V(I,N)
      AL1C = AR1*AU1 - ALOC
      AL1Y = AR1*AU1*AV1 - ALOY
      AL2C = R(J,N)*U(J,N) -ALOC
      AL2Y = R(J,N)*U(J,N)*V(J,N) -ALOY
      A1C = (AL1C*Y20SQ-AL2C*Y10SQ)/D
      A1Y = (AL1Y*Y20SQ-AL2Y*Y10SQ)/D
      A2C = (AL2C*Y10 - AL1C*Y20)/D
      A2Y = (AL2Y*Y10 - AL1Y*Y20)/D
      F1  = Y10*(D-EY2*Y10/2.+Y10SQ*Y21/3.)
      F2 = Y10SQ*Y20*(Y20/2.-Y10/3.)
      FC1 = AL1C+2.*AL2C
      FC2 = Y20 -Y10/3.
      FC3 = AL1Y+2.*AL2Y
      FC4 = Y10/2.-Y20/3.
      FC5 = AL1C-AL2C
      FC6 = Y10-Y20/3.
      FC7 = A1C/2.+Y20*A2C/3.
      FC8 = AL1Y-AL2Y
      FC9 = A1Y/2.+Y20*A2Y/3.
```

```
      F3C = -Y10QU*DRHU(J)/6.
      F3C = F3C +(-AL1C*D+Y10SQ*(Y10*FC1/3.-Y20*AL1C))*DY
      F3C = F3C -2.*AL2C*Y10SQ*Y10/3.*DY1
      F3C = F3C +AL1C*Y10SQ*FC2*DY2
      F3C = F3C -Y10SQ*(A1C/2.+Y10*A2C/3.)*DD
      F3C = F3C +(AR1*AV1+R(I,N)*V(I,N))*D
      F3Y = -Y10QU*DRUV(J)/6.
      F3Y = F3Y +(-AL1Y*D +Y10SQ*(Y10*FC3/3.-Y20*AL1Y))*DY
      F3Y = F3Y -2.*AL2Y*Y10SQ*Y10/3.*DY1
      F3Y = F3Y +AL1Y*Y10SQ*FC2*DY2
      F3Y = F3Y -Y10SQ*(A1Y/2.+Y10*A2Y/3.)*DD
      F3Y = F3Y +(CK*AP1+AR1*AV1*AV1-CK*P(I,N)-R(I,N)*V(I,N)*V(I,N))*D
      F4 = Y20*(D + Y20*(-EY2/2. + Y20*Y21/3.))
      F5  = Y20QU/6.
      F6C = -Y10*Y20SQ*FC4*DRHU(J)
      F6C = F6C +(-AL2C*D+Y20SQ*(AL2C*Y10-AL1C*Y20+Y20*FC5/3.))*DY
      F6C = F6C -Y20SQ*(AL2C*FC6*DY1-2.*Y20*AL1C/3.*DY2+FC7*DD)
      F6C = F6C +(R(J,N)*V(J,N)+R(I,N)*V(I,N))*D
      F6Y = -Y10*Y20SQ*FC4*DRUV(J)
      F6Y = F6Y +(-AL2Y*D +Y20SQ*(AL2Y*Y10-AL1Y*Y20+Y20*FC8/3.))*DY
      F6Y = F6Y -Y20SQ*(AL2Y*FC6*DY1 -2.*Y20*AL1Y/3.*DY2+FC9*DD)
      F6Y = F6Y +(CK*(P(J,N)-P(I,N)) +R(J,N)*V(J,N)*V(J,N) -R(I,N)*
     1     V(I,N)*V(I,N))*D
      DEL = F1*F5 - F2*F4
  400 E2 = F4*F3C - F1*F6C
      E6 = F4*F3Y - F1*F6Y
      ZVS(K) = DVDY/(AVS**2*AR1/(1.4*CK*AP1)-1.)
      ZV1(K) = (E6-AV1*E2)/(AR1*AU1*DEL)
  700 ZY1(K) = DY1
      DVS    = (ZVS(2) + 2.*(ZVS(3) + ZVS(4)) + ZVS(5))/6.
      DV1    = (ZV1(2) + 2.*(ZV1(3) + ZV1(4)) + ZV1(5))/6.
      DY1    = (ZY1(2) + 2.*(ZY1(3) + ZY1(4)) + ZY1(5))/6.
      RETURN
      END
```

```
      SUBROUTINE OUNS(M,I)
C
C   THIS SUBROUTINE PERFORMS A FLOW INTEGRATION STEP ON THE NEXT TO THE
C   OUTERMOST STREAMLINE
C
      COMMON       C        ,CK       ,RS       ,FM       ,ALPHA
      COMMON/ACOM/X         ,XA       ,VN       ,VS       ,H        ,DY
     1    ,YO      ,PO      ,RO       ,UO       ,VO       ,RMO      ,DUO
     2    ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/OCOM/          DRHU(2),DPRU(2),DRUV(2)
      DIMENSION             ZU1(5),ZV1(5),ZY1(5), ZRHU(5),ZPRU(5),ZRUV(5)
      DATA ZU1,ZV1,ZY1/ 15*0.0    /
      Y20 = Y(I,1) -YO*M
      Y20SQ = Y20*Y20
      Y20QU = Y20SQ*Y20SQ
      DY = M*VO/UO
      ALOC =   RO* UO
      ALOX = CK* PO + ALOC* UO
      ALOY = M*ALOC*VO
      AL2C = 1. - ALOC
      AL2X = CK    + 1.        - ALOX
      AL2Y =              - ALOY
      DO 700 K=2,5
      RN =(K-1)/2
      RNH = RN/2.*H
      AY1 = Y(I,2)+ZY1(K-1)*RNH
      AU1 = U(I,2)+ZU1(K-1)*RNH
      AV1 = V(I,2)+ZV1(K-1)*RNH
      IF     (C - AU1*AU1 - AV1*AV1) 100, 300, 300
  100 X = -1.
      RETURN
  300 AR1 = ((C - AU1*AU1 - AV1*AV1)/(C - 1.))**2.5
      AP1 = AR1**1.4
      Y10 = AY1 -YO*M
      Y21 = Y(I,1)-AY1
      Y10SQ = Y10*Y10
      DY1 = AV1/AU1
      D = Y10*Y20*Y21
      DD = -Y21*(Y20+Y10)*DY + Y20*(Y21-Y10)*DY1
      EY2 = Y20SQ-Y10SQ
      AL1C = AR1*AU1 - ALOC
      AL1X = CK*AP1 + AR1*AU1*AU1 - ALOX
      AL1Y = AR1*AU1*AV1 - ALOY
      A1C = (AL1C*Y20SQ-AL2C*Y10SQ)/D
      A1X = (AL1X*Y20SQ-AL2X*Y10SQ)/D
      A1Y = (AL1Y*Y20SQ-AL2Y*Y10SQ)/D
      A2C = (AL2C*Y10 - AL1C*Y20)/D
      A2X = (AL2X*Y10 - AL1X*Y20)/D
      A2Y = (AL2Y*Y10 - AL1Y*Y20)/D
      F1  = Y10*(D-EY2*Y10/2.+Y10SQ*Y21/3.)
      F2 = Y10SQ*Y20*(Y20/2.-Y10/3.)
      FC1 = AL1C+2.*AL2C
      FC3 = AL1Y+2.*AL2Y
      FC5 = AL1C-AL2C
      FC6 = Y10-Y20/3.
      FC7 = A1C/2.+Y20*A2C/3.
      FC8 = AL1Y-AL2Y
      FC9 = A1Y/2.+Y20*A2Y/3.
      FCA = AL1X+2.*AL2X
      FCB = A1X/2.+Y10*A2X/3.
      FCC = AL1X-AL2X
      FCD = A1X/2.+Y20*A2X/3.
      F3C = (-AL1C*D+Y10SQ*(Y10*FC1/3.-Y20*AL1C))*DY
      F3C = F3C -2.*AL2C*Y10SQ*Y10/3.*DY1
      F3C = F3C -Y10SQ*(A1C/2.+Y10*A2C/3.)*DD
      F3C = F3C +(AR1*AV1-RO*VO*M)*D
      F3X = (-AL1X*D+Y10SQ*(Y10*FCA/3.-Y20*AL1X))*DY
      F3X = F3X -2.*AL2X*Y10SQ*Y10/3.*DY1
      F3X = F3X -Y10SQ*FC3*DD+AL1Y*D
      F3Y = (-AL1Y*D+Y10SQ*(Y10*FC3/3.-Y20*AL1Y))*DY
```

```
      F3Y = F3Y -2.*AL2Y*Y10SQ*Y10/3.*DY1
      F3Y = F3Y -Y10SQ*(A1Y/2.+Y10*A2Y/3.)*DD
      F3Y = F3Y +(CK*AP1+AR1*AV1*AV1-CK*PO-RO*VO*VO)*D
      F4 = Y20*(D + Y20*(-EY2/2. + Y20*Y21/3.))
      F5  = Y20QU/6.
      F6C = (-AL2C*D+Y20SQ*(AL2C*Y10-AL1C*Y20+Y20*FC5/3.))*DY
      F6C = F6C -Y20SQ*(AL2C*FC6*DY1+FC7*DD)
      F6C = F6C -RO*VO*M*D
      F6X = (-AL2X*D+Y20SQ*(AL2X*Y10-AL1X*Y20+Y20*FCC/3.))*DY
      F6X = F6X -Y20SQ*(AL2X*FC6*DY1+FC0*DD)+AL2Y*D
      F6Y = (-AL2Y*D+Y20SQ*(AL2Y*Y10-AL1Y*Y20+Y20*FC8/3.))*DY
      F6Y = F6Y -Y20SQ*(AL2Y*FC6*DY1+FC9*DD)
      F6Y = F6Y +(CK*(1.-PO) -RO*VO*VO)*D
      DEL = F1*F5 - F2*F4
      E2 = F4*F3C - F1*F6C
      E4 = F4*F3X - F1*F6X
      E6 = F4*F3Y - F1*F6Y
      CO1 = DEL*(1.4*CK*AP1/AR1 - AU1*AU1)
      ZU1(K) = ((CK*1.4*AP1/AR1 +AU1*AU1)*E2 -AU1*E4)/(AR1*CO1)
      ZV1(K) = (E6 -AV1*E2)/(AR1*AU1*DEL)
      ZRHU(K)= E2/DEL
      ZPRU(K)= E4/DEL
      ZRUV(K)= E6/DEL
  700 ZY1(K) = DY1
      DV1    = (ZV1(2) + 2.*(ZV1(3) + ZV1(4)) + ZV1(5))/6.
      DY1    = (ZY1(2) + 2.*(ZY1(3) + ZY1(4)) + ZY1(5))/6.
      DU(I,2)    = (ZU1(2)+2.*(ZU1(3)+ZU1(4))+ZU1(5))/6.
      DRHU(I)= (ZRHU(2)+2.*(ZRHU(3)+ZRHU(4))+ZRHU(5))/6.
      DPRU(I)= (ZPRU(2)+2.*(ZPRU(3)+ZPRU(4))+ZPRU(5))/6.
      DRUV(I)= (ZRUV(2)+2.*(ZRUV(3)+ZRUV(4))+ZRUV(5))/6.
      U(I,2) = U(I,2) +H*DU(I,2)
      V(I,2) = V(I,2) +H*DV1
      Y(I,2) = Y(I,2) +H*DY1
      VSQ = U(I,2)*U(I,2)+V(I,2)*V(I,2)
      R(I,2) = ((C-VSQ)/(C-1.))**2.5
      P(I,2) = R(I,2)**1.4
      RM(I,2)= SQRT(VSQ*R(I,2)/(1.4*CK*P(I,2)))
      RETURN
      END
```

```
      SUBROUTINE INAS(M,I,N,IJ)
C
C   THIS SUBROUTINE PERFORMS A FLOW INTEGRATION STEP ON THE NTH STRIP
C   IN SOME CASES THIS SUBROUTINE PERFORMS A FLOW INTEGRATION STEP ON
C   THE STAGNATION STREAMLINE
C
      COMMON       C        ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X         ,XA      ,VN      ,VS      ,H       ,DY
     1   ,YO      ,PO       ,RO      ,UO      ,VO      ,RMO     ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/DCOM/CS        ,CZ      ,DV1     ,Q1      ,DQ1     ,RK
     1   ,VOO     ,ISKIP
      COMMON/OCOM/          DRHU(2),DPRU(2),DRUV(2)
      DIMENSION         ZU1(5),ZV1(5),ZY1(5), ZRHU(5),ZPRU(5),ZRUV(5)
      DIMENSION ZUO(5),ZVO(5)
      DATA ZUO,E1,E3,E5/8*0.0/
      DATA ZU1,ZV1,ZY1/ 15*0.0   /
      Y20 = Y(I,N-1)-YO*M
      Y20SQ = Y20*Y20
      Y20QU = Y20SQ*Y20SQ
      DO 700 K=2,5
      RN = (K-1)/2
      RNH = RN/2.*H
      IF(N.EQ.IJ) GO TO 40
      AUO = UO
      AVO = VO*M
      ARO = RO
      APO = PO
      GO TO 220
   40 AUO = UO+ZUO(K-1)*RNH
      GO TO (50,60,80),ISKIP
   50 AVO = VOO*EXP((1.-X)*RK)
      GO TO 90
   60 AVO = AUO*DY
      GO TO 90
   80 CONTINUE
      AVO = VO +ZVO(K-1)*RNH
   90 CONTINUE
      IF(C-AUO*AUO-AVO*AVO) 100,180,180
  100 X = -1.
      RETURN
  180 ARO = ((CZ-AUO*AUO-AVO*AVO)/(CS*(C-1.)))**2.5
      APO = ARO**1.4*CS
  220 AY1 = Y(I,N) +ZY1(K-1)*RNH
      AU1 = U(I,N)+ZU1(K-1)*RNH
      AV1 = V(I,N)+ZV1(K-1)*RNH
      IF     (C - AU1*AU1 - AV1*AV1) 100, 300, 300
  300 AR1 = ((C - AU1*AU1 - AV1*AV1)/(C - 1.))**2.5
      AP1 = AR1**1.4
      Y10 = AY1-YO*M
      Y21 = Y(I,N-1)-AY1
      Y10SQ = Y10*Y10
      Y10QU = Y10SQ*Y10SQ
      DY = AVO/AUO
      DY1 = AV1/AU1
      DY2 = V(I,N-1)/U(I,N-1)
      D = Y10*Y20*Y21
      DD = -Y21*(Y20+Y10)*DY + Y20*(Y21-Y10)*DY1    +Y10*(Y21+Y20)*DY2
      EY2 = Y20SQ-Y10SQ
      ALOC =ARO*AUO
      ALOX =CK*APO+ALOC*AUO
      ALOY = ALOC*AVO
      AL1C = AR1*AU1 - ALOC
      AL1X = CK*AP1 + AR1*AU1*AU1 - ALOX
      AL1Y = AR1*AU1*AV1 - ALOY
      AL2C= R(I,N-1)*U(I,N-1) -ALOC
      AL2X= CK*P(I,N-1) +R(I,N-1)*U(I,N-1)*U(I,N-1) -ALOX
      AL2Y= R(I,N-1)*U(I,N-1)*V(I,N-1) -ALOY
      A1C = (AL1C*Y20SQ-AL2C*Y10SQ)/D
      A1X = (AL1X*Y20SQ-AL2X*Y10SQ)/D
```

```
      A1Y = (AL1Y*Y20SQ-AL2Y*Y10SQ)/D
      A2C = (AL2C*Y10 - AL1C*Y20)/D
      A2X = (AL2X*Y10 - AL1X*Y20)/D
      A2Y = (AL2Y*Y10 - AL1Y*Y20)/D
      F1  = Y10*(D-EY2*Y10/2.+Y10SQ*Y21/3.)
      F2 = Y10SQ*Y20*(Y20/2.-Y10/3.)
      FC1 = AL1C+2.*AL2C
      FC2 = Y20 -Y10/3.
      FC3 = AL1Y+2.*AL2Y
      FC4 = Y10/2.-Y20/3.
      FC5 = AL1C-AL2C
      FC6 = Y10-Y20/3.
      FC7 = A1C/2.+Y20*A2C/3.
      FC8 = AL1Y-AL2Y
      FC9 = A1Y/2.+Y20*A2Y/3.
      FCA = AL1X+2.*AL2X
      FCB = A1X/2.+Y10*A2X/3.
      FCC = AL1X-AL2X
      FCD = A1X/2.+Y20*A2X/3.
      F3C = -Y10QU*DRHU(I)/6.
      F3C = F3C +(-AL1C*D+Y10SQ*(Y10*FC1/3.-Y20*AL1C))*DY
      F3C = F3C -2.*AL2C*Y10SQ*Y10/3.*DY1
      F3C = F3C +AL1C*Y10SQ*FC2*DY2
      F3C = F3C -Y10SQ*(A1C/2.+Y10*A2C/3.)*DD
      F3C = F3C+(AR1*AV1-AR0*AV0)*D
      F3X = -Y10QU*DPRU(I)/6.
      F3X = F3X +(-AL1X*D+Y10SQ*(Y10*FCA/3.-Y20*AL1X))*DY
      F3X = F3X -2.*AL2X*Y10SQ*Y10/3.*DY1
      F3X = F3X +AL1X*Y10SQ*FC2*DY2
      F3X = F3X -Y10SQ*FCB*DD+AL1Y*D
      F3Y = -Y10QU*DRUV(I)/6.
      F3Y = F3Y +(-AL1Y*D +Y10SQ*(Y10*FC3/3.-Y20*AL1Y))*DY
      F3Y = F3Y -2.*AL2Y*Y10SQ*Y10/3.*DY1
      F3Y = F3Y +AL1Y*Y10SQ*FC2*DY2
      F3Y = F3Y -Y10SQ*(A1Y/2.+Y10*A2Y/3.)*DD
      F3Y = F3Y+(CK*AP1+AR1*AV1*AV1-CK*AP0-AR0*AV0*AV0)*D
      F4 = Y20*(D + Y20*(-EY2/2. + Y20*Y21/3.))
      F5  = Y20QU/6.
      F6C = -Y10*Y20SQ*FC4*DRHU(I)
      F6C = F6C +(-AL2C*D+Y20SQ*(AL2C*Y10-AL1C*Y20+Y20*FC5/3.))*DY
      F6C = F6C -Y20SQ*(AL2C*FC6*DY1-2.*Y20*AL1C/3.*DY2+FC7*DD)
      F6C = F6C+(R(I,N-1)*V(I,N-1)-AR0*AV0)*D
      F6X = -Y10*Y20SQ*FC4*DPRU(I)
      F6X = F6X +(-AL2X*D+Y20SQ*(AL2X*Y10-AL1X*Y20+Y20*FCC/3.))*DY
      F6X = F6X-Y20SQ*(AL2X*FC6*DY1-2.*Y20*AL1X/3.*DY2+FCD*DD)+AL2Y*D
      F6Y = -Y10*Y20SQ*FC4*DRUV(I)
      F6Y = F6Y +(-AL2Y*D +Y20SQ*(AL2Y*Y10-AL1Y*Y20+Y20*FC8/3.))*DY
      F6Y = F6Y -Y20SQ*(AL2Y*FC6*DY1 -2.*Y20*AL1Y/3.*DY2+FC9*DD)
      F6Y =F6Y+(CK*(P(I,N-1)-AP0)+R(I,N-1)*V(I,N-1)*V(I,N-1)
     1-AR0*AV0*AV0)*D
      DEL = F1*F5 - F2*F4
      IF(N.LT.IJ.OR.IJ.LE.1) GO TO 400
  380 E1 = F2*F6C-F5*F3C
      E3 = F2*F6X-F5*F3X
      E5 = F2*F6Y-F5*F3Y
      ZV0(K) = (E5-AV0*E1)/(AR0*AU0*DEL)
      CD0 = DEL*(1.4*CK*AP0/AR0-AU0*AU0)
      ZU0(K) = ((CK*1.4*AP0/AR0+AU0*AU0)*E1-AU0*E3)/(AR0*CD0)
  400 E2 = F4*F3C - F1*F6C
      E4 = F4*F3X - F1*F6X
      E6 = F4*F3Y - F1*F6Y
      CD1 = DEL*(1.4*CK*AP1/AR1 - AU1*AU1)
      ZRHU(K)= E2/DEL
      ZPRU(K)= E4/DEL
      ZRUV(K)= E6/DEL
      ZU1(K) = ((CK*1.4*AP1/AR1 +AU1*AU1)*E2 -AU1*E4)/(AR1*CD1)
      ZV1(K) = (E6-AV1*E2)/(AR1*AU1*DEL)
  700 ZY1(K) = DY1
      DV1    = (ZV1(2) + 2.*(ZV1(3) + ZV1(4)) + ZV1(5))/6.
      DY1    = (ZY1(2) + 2.*(ZY1(3) + ZY1(4)) + ZY1(5))/6.
```

```
      DU(I,N)     = (ZU1(2)+2.*(ZU1(3)+ZU1(4))+ZU1(5))/6.
      IF(IJ.LE.1) GO TO 704
      IF(ISKIP.EQ.1) GO TO 710
  704 DRHU(I)= (ZRHU(2)+2.*(ZRHU(3)+ZRHU(4))+ZRHJ(5))/6.
      DPRU(I)= (ZPRU(2)+2.*(ZPRU(3)+ZPRU(4))+ZPRJ(5))/6.
      DRUV(I)= (ZRUV(2)+2.*(ZRUV(3)+ZRUV(4))+ZRUV(5))/6.
      IF(N.LT.IJ.OR.IJ.LE.1) GO TO 720
      DO 705 K=1,25
      XA = XA+H*0.05
      CALL ARFL(XA,XB,YO,DY,DUM,I)
      IF(XB.GT.X) GO TO 710
  705 CONTINUE
  710 DUO = (ZUO(2)+2.*(ZUO(3)+ZUO(4))+ZUO(5))/6.
      UO = UO+H*DUO
      GO TO (714,715,716),ISKIP
  714 VO = VOO*EXP((1.-X)*RK)
      YO = YO+H*VO/UO
      GO TO 718
  715 VO = UO*DY
      GO TO 718
  716 DVO = (ZVO(2)+2.*(ZVO(3)+ZVO(4))+ZVO(5))/6.
      VO = VO+H*DVO
      VSQ = UO*UO+VO*VO
      UO = SQRT(VSQ/(1.+DY*DY))
      VO = UO*DY
  718 VSQ = UO*UO+VO*VO
      RO = ((CZ-VSQ)/(CS*(C-1.)))**2.5
      PO = CS*RO**1.4
      RMO = SQRT(RO*VSQ/(1.4*CK*PO))
  720 U(I,N) = U(I,N)+H*DU(I,N)
      V(I,N) = V(I,N)+H*DV1
      Y(I,N) = Y(I,N)+H*DY1
      VSQ = U(I,N)*U(I,N)+V(I,N)*V(I,N)
      R(I,N) = ((C-VSQ)/(C-1.))**2.5
      P(I,N) = R(I,N)**1.4
      RM(I,N)= SQRT(VSQ*R(I,N)/(1.4*CK*P(I,N)))
      Q1 = SQRT(VSQ)
      DQ1 = (U(I,N)*DU(I,N)+V(I,N)*DV1)/Q1
      RETURN
      END
```

```
      SUBROUTINE INBO(N,I)
C
C   THIS SUBROUTINE PERFORMS A FLOW INTEGRATION STEP ALONG THE
C   STAGNATION STREAMLINE
C
      COMMON     C       ,CK      ,RS      ,FM      ,ALPHA
      COMMON/ACOM/X      ,XA      ,VN      ,VS      ,H       ,DY
     1   ,YO     ,PO      ,RO      ,UO      ,VO      ,RMO     ,DUO
     2   ,Y(2,10),P(2,10),R(2,10),U(2,10),V(2,10),RM(2,10),DU(2,10)
      COMMON/CCOM/XB      ,YB      ,DYB     ,DDYB    ,DUB     ,PB
     1   ,UB     ,RMB     ,DB      ,HS      ,CRA
      COMMON/OCOM/        DRHU(2),DPRU(2),DRUV(2)
      DIMENSION ZU1(5), ZV1(5), ZY1(5), ZVS(5), ZVN(5), ZB (5), ZUB(5)
      DATA ZU1,ZV1,ZVS,ZVN,ZUB,ZB,ZY1/35*0.0/
      CT = 1./SQRT(1. + DYB**2)
      ST = DYB*CT
      RABO= ABS(1./(CT**3*DDYB))
      DO 700 K = 2, 5
      RN=1.0
      IF(K.EQ.2) RN=0.0
      IF(K.EQ.5) RN=2.0
      RNH = RN/2.*H
      AY1 = Y(I,N)+ZY1(K-1)*RNH
      AU1 = U(I,N)+ZU1(K-1)*RNH
      AV1 = V(I,N)+ZV1(K-1)*RNH
      IF (     C - AU1*AU1 - AV1*AV1) 100, 300, 300
  100 RMB = 2.0
      RETURN
  300 AR1 = ((C - AU1*AU1 - AV1*AV1)/(C - 1.))**2.5
      AP1 = AR1**1.4
      AVS = VS +ZVS(K-1)*RNH*HS/H
      AVN = VN +ZVN(K-1)*RNH*HS/H
      AUO = AVS*CT - AVN*ST
      AVO = AVS*ST + AVN*CT
      IF (C - AUO**2 - AVO**2) 100, 400, 400
  400 ARO = ((C - AUO*AUO - AVO*AVO)/(C - 1.))**2.5
      APO = ARO**1.4
      AUB = UB +ZUB(K-1)*RNH*HS/H
      IF (C - AUB**2) 100, 500, 500
  500 ARB = ((C - AUB*AUB              )/(C - 1.))**2.5
      APB = ARB**1.4
      B   = DB +ZB(K-1)*RNH*HS/H
      RAB = RABO+B*CRA
      AYO = YB +B*CT
      Y10 = AY1 -AYO
      Y20 = Y(I,N-1)-AYO
      Y21 = Y(I,N-1)-AY1
      DY = AVO/AUO
      DY1 = AV1/AU1
      DY2 = V(I,N-1)/U(I,N-1)
      D = Y10*Y20*Y21
      Y10SQ = Y10*Y10
      Y20SQ = Y20*Y20
      Y10CU = Y10SQ*Y10SQ
      Y20CU = Y20SQ*Y20SQ
      DD = -Y21*(Y20+Y10)*DY + Y20*(Y21-Y10)*DY1    +Y10*(Y21+Y20)*DY2
      EY2 = Y20SQ-Y10SQ
      ALOC = ARO*AUO
      ALOX = CK*APO + ALOC*AUO
      ALOY = ALOC*AVO
      AL1C = AR1*AU1 - ALOC
      AL1X = CK*AP1 + AR1*AU1*AU1 - ALOX
      AL1Y = AR1*AU1*AV1 - ALOY
          AL2C = R(I,N-1)*U(I,N-1) -ALOC
      AL2X = CK*P(I,N-1) +R(I,N-1)*U(I,N-1)*U(I,N-1) -ALOX
      AL2Y = R(I,N-1)*U(I,N-1)*V(I,N-1) -ALOY
      A1C = (AL1C*Y20SQ-AL2C*Y10SQ)/D
      A1X = (AL1X*Y20SQ-AL2X*Y10SQ)/D
      A1Y = (AL1Y*Y20SQ-AL2Y*Y10SQ)/D
      A2C = (AL2C*Y10 - AL1C*Y20)/D
```

```
      A2X = (AL2X*Y10 - AL1X*Y20)/D
      A2Y = (AL2Y*Y10 - AL1Y*Y20)/D
      F1  = Y10*(D-EY2*Y10/2.+Y10SQ*Y21/3.)
      F2 = Y10SQ*Y20*(Y20/2.-Y10/3.)
      FC1 = AL1C+2.*AL2C
      FC2 = Y20 -Y10/3.
      FC3 = AL1Y+2.*AL2Y
      FC4 = Y10/2.-Y20/3.
      FC5 = AL1C-AL2C
      FC6 = Y10-Y20/3.
      FC7 = A1C/2.+Y20*A2C/3.
      FC8 = AL1Y-AL2Y
      FC9 = A1Y/2.+Y20*A2Y/3.
      FCA = AL1X+2.*AL2X
      FCB = A1X/2.+Y10*A2X/3.
      FCC = AL1X-AL2X
      FCD = A1X/2.+Y20*A2X/3.
      F3C = -Y10QU*DRHU(I)/6.
      F3C = F3C +(-AL1C*D+Y10SQ*(Y10*FC1/3.-Y20*AL1C))*DY
      F3C = F3C -2.*AL2C*Y10SQ*Y10/3.*DY1
      F3C = F3C +AL1C*Y10SQ*FC2*DY2
      F3C = F3C -Y10SQ*(A1C/2.+Y10*A2C/3.)*DD
      F3C = F3C +(AR1*AV1-AR0*AV0)*D
      F3X = -Y10QU*DPRU(I)/6.
      F3X = F3X +(-AL1X*D+Y10SQ*(Y10*FCA/3.-Y20*AL1X))*DY
      F3X = F3X -2.*AL2X*Y10SQ*Y10/3.*DY1
      F3X = F3X +AL1X*Y10SQ*FC2*DY2
      F3X = F3X -Y10SQ*FCB*DD+AL1Y*D
      F3Y = -Y10QU*DRUV(I)/6.
      F3Y = F3Y +(-AL1Y*D +Y10SQ*(Y10*FC3/3.-Y20*AL1Y))*DY
      F3Y = F3Y -2.*AL2Y*Y10SQ*Y10/3.*DY1
      F3Y = F3Y +AL1Y*Y10SQ*FC2*DY2
      F3Y = F3Y -Y10SQ*(A1Y/2.+Y10*A2Y/3.)*DD
      F3Y = F3Y +(CK*AP1+AR1*AV1*AV1-CK*AP0-AR0*AV0*AV0)*D
      F4 = Y20*(D + Y20*(-EY2/2. + Y20*Y21/3.))
      F5  = Y20QU/6.
      F6C = -Y10*Y20SQ*FC4*DRHU(I)
      F6C = F6C +(-AL2C*D+Y20SQ*(AL2C*Y10-AL1C*Y20+Y20*FC5/3.))*DY
      F6C = F6C -Y20SQ*(AL2C*FC6*DY1-2.*Y20*AL1C/3.*DY2+FC7*DD)
      F6C = F6C +(R(I,N-1)*V(I,N-1)-AR0*AV0)*D
      F6X = -Y10*Y20SQ*FC4*DPRU(I)
      F6X = F6X +(-AL2X*D+Y20SQ*(AL2X*Y10-AL1X*Y20+Y20*FCC/3.))*DY
      F6X = F6X-Y20SQ*(AL2X*FC6*DY1-2.*Y20*AL1X/3.*DY2+FCD*DD)+AL2Y*D
      F6Y = -Y10*Y20SQ*FC4*DRUV(I)
      F6Y = F6Y +(-AL2Y*D +Y20SQ*(AL2Y*Y10-AL1Y*Y20+Y20*FC8/3.))*DY
      F6Y = F6Y -Y20SQ*(AL2Y*FC6*DY1 -2.*Y20*AL1Y/3.*DY2+FC9*DD)
      F6Y = F6Y+(CK*(P(I,N-1)-AP0)+R(I,N-1)*V(I,N-1)*V(I,N-1)
     1-AR0*AV0*AV0)*D
      DEL = F1*F5 - F2*F4
      E1 = F2*F6C - F5*F3C
      E2 = F4*F3C - F1*F6C
      E3 = F2*F6X - F5*F3X
      E4 = F4*F3X - F1*F6X
      E5 = F2*F6Y - F5*F3Y
      E6 = F4*F3Y - F1*F6Y
      C00 = DEL*(1.4*CK*AP0/AR0 - AU0*AU0)
      C01 = DEL*(1.4*CK*AP1/AR1 - AU1*AU1)
      ZY1(K) = DY1
      ZU1(K) = ((CK*1.4*AP1/AR1 +AU1*AU1)*E2 -AU1*E4)/(AR1*C01)
      ZV1(K) = (E6-AV1*E2)/(AR1*AU1*DEL)
      RABPB = RAB+B
      ZB(K)  = (1.+B/ RABPB )*AVN/AVS
      DU0    = ((CK*1.4*AP0/AR0 + AU0*AU0)*E1 - AU0*E3)/(AR0*C00)
      DV0 = (E5 - AV0*E1)/(AR0*AU0*DEL)
      ZVS(K) = (DU0*CT+DV0*ST)*CT-AVN/RAB
      OMS = 1. - AVS**2*AR0/(1.4*CK*AP0)
      FF =  (2./B +1./RAB)*CK*APB
      FF = FF +ARB*AUB*AUB/RAB
      FF = FF -(2./B +1./ RABPB )*CK*AP0
      FF = FF +AR0*AVS*AVS/ RABPB
```

```
      FF = FF -2.*(1./B+1./RABPB)*ARO*AVN*AVN
      FF = FF/ARO/AVS
      ZVN(K) =  AVN/ B*ZB(K)-OMS*AVN/AVS*ZVS(K) +FF
      ZVN(K) = ZVN(K)/(1.-AVN*AVS*ARO/(1.4*CK*APO))
      DRVS   =  ARO*(OMS*ZVS(K)-ARO*AVS*AVN/(1.4*CK*APO)*ZVN(K))
      ZUB(K) = (ARB*AUB +ARO*AVS)*ZB(K)/B +DRVS
      ZUB(K) = ZUB(K)/(ARB*(AUB*AUB/(1.4*CK*APB/ARB) -1.))
  700 CONTINUE
      DY1    = (ZY1(2) + 2.*(ZY1(3) + ZY1(4)) + ZY1(5))/6.
      DU(I,N)= (ZU1(2)+2.*(ZU1(3)+ZU1(4))+ZU1(5))/6.
      DV1    = (ZV1(2) + 2.*(ZV1(3) + ZV1(4)) + ZV1(5))/6.
      DVS    = (ZVS(2) + 2.*(ZVS(3) + ZVS(4)) + ZVS(5))/6.
      DVN    = (ZVN(2) + 2.*(ZVN(3) + ZVN(4)) + ZVN(5))/6.
      DUB    = (ZUB(2) + 2.*(ZUB(3) + ZUB(4)) + ZUB(5))/6.
      DDB    = (ZB (2) + 2.*(ZB (3) + ZB (4)) + ZB (5))/6.
      Y(I,N) = Y(I,N)+H*DY1
      U(I,N) = U(I,N)+H*DU(I,N)
      V(I,N) = V(I,N)+H*DV1
      VSQ    =  U(I,N)*U(I,N)+V(I,N)*V(I,N)
      R(I,N) =  ((C-VSQ)/(C-1.))**2.5
      P(I,N) = R(I,N)**1.4
      RM(I,N) =  SQRT(VSQ*R(I,N)/(1.4*CK*P(I,N)))
      DB = DB +DDB*HS
      VS = VS + DVS*HS
      VN = VN + DVN*HS
      UB = UB +   DUB*HS
      RB = ((C - UB*UB)/(C - 1.))**2.5
      PB = RB**1.4
      RMB = UB/SQRT(1.4*CK*PB/RB)
      XA = XA + HS*CT
      CALL ARFL ( XA,XB, YB, DYB, DDYB,I)
      CT = 1./SQRT(1. + DYB**2)
      ST = CT*DYB
      H = XB -DB*ST-X
      X = X + H
      YO = YB +DB*CT
      UO = VS*CT - VN*ST
      VO = VS*ST + VN*CT
      RO  = ((C -  UO* UO -  VO* VO)/(C - 1.))**2.5
      PO = RO**1.4
      RMO = SQRT((UO*UO + VO*VO)*RO/(1.4*CK* PO))
      RETURN
      END
```

APPENDIX G

IGS PROGRAM LISTING

```
      BLOCK DATA CGRAF                                                  CGRAF      2
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NM01B(6) ,NM02B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NCHARS/NNEQ     ,NAEQ     ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2)    NCHARS     2
     1   ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)    NCHARS     3
     2   ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)   ,NMACHQ(2),NALPHA(2)   NCHARS     4
     3   ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI     ,FMTF                  NCHARS     5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
C                                                                       CGRAF      9
C    THESE DATA STATEMENTS CONTAIN CHARACTERS USED IN THE CREATION OF THE CGRAF   10
C    LIGHT REGISTERS AND MANY TEXT ENTITIES                             CGRAF     11
      DATA NNEQ,NAEQ/  10HNN=       ,10HNA=       /                     CGRAF     12
      DATA NDVIEQ,NXSEQ,NYSOEQ,NXAEQ,NCYDEQ,NSLEQ,NXOOEQ/               CGRAF     13
     1    10HDVOO(I)=  ,10H         ,10HXS=       ,10H          ,       CGRAF     14
     2    10HYSO=      ,10H         ,10HXA=       ,10H          ,       CGRAF     15
     3    10HCYD=      ,10H         ,10HSHOCK L=  ,10H          ,       CGRAF     16
     4    10HXOO=      ,10H         /                                   CGRAF     17
      DATA NMACHQ,NALPHA,NYIUEQ,NYILEQ/                                 CGRAF     18
     1    10HMACH NO.= ,10H         ,10HALPHA=    ,10H          ,       CGRAF     19
     2    10HYI(UPR)=  ,10H         ,10HYI(LWR)=  ,10H          /       CGRAF     20
      DATA NDEEQ,NYSEQ,NDVOEQ,NRUEQ,NUBEQ,NID,NPOEQ/                    CGRAF     21
     1    10HDE=       ,10H         ,10HYS=       ,10H          ,       CGRAF     22
     2    10HDVOO(F)=  ,10H         ,10HRBUB=     ,10H          ,       CGRAF     23
     3    10HUB=       ,10H         ,10HFLOWS NOT ,10HMATCHED   ,       CGRAF     24
     4    10HPO=       ,10H         /                                   CGRAF     25
      DATA FMTI,FMTF/  4H(I1)       ,7H(F10.8)    /                     CGRAF     26
C                                                                       CGRAF     27
C    THIS DATA STATEMENT CONTAINS THE SIX INTEGER ARRAYS IDENTIFYING THE CGRAF    28
C    TEXT ENTITIES DISPLAYING SPECIFIED OUTPUT VALUES IN THE LOWER RIGHT CGRAF    29
C    HAND CORNER OF THE SCREEN                                          CGRAF     30
      DATA LRXOOQ  ,LRDIEQ   ,LRXSEQ   ,LRXAUP   ,LRCYDU   ,LRXALW   ,  CGRAF     31
     1    LRCYDL   ,LRSLEQ   ,LRMACH   ,LRALFA   ,LRYIU    ,LRYIL    ,  CGRAF     32
     2    LRNN1    ,LRNA2    ,LRNN3    ,LRNN4    ,LRNN5    ,LRNN6    ,  CGRAF     33
     3    LRUPS    ,LRSTG    ,LRAFU2   ,LRAFL2   ,LRAFU3   ,LRSTRT   /  CGRAF     34
     5    1,1,4*99 ,2,2,4*99 ,3,2,4*99 ,4,3,4*99 ,5,3,4*99 ,6,4,4*99 ,  CGRAF     35
     6    7,4,4*99 ,8,5,4*99 ,11,6,4*99,12,6,4*99,13,6,4*99,14,6,4*99,  CGRAF     36
     7    21,1,4*99,22,1,4*99,23,3,4*99,24,4,4*99,25,5,4*99,26,5*99  ,  CGRAF     37
     8    0,1,4*0  ,0,2,4*0  ,0,3,4*0  ,0,4,4*0  ,0,5,4*0  ,0,6,4*0  /  CGRAF     38
      DATA LRDEEQ  ,LRYSOQ   ,LRYSEQ   ,LRDOEQ   ,LRRUEQ   ,LRUBEQ   ,  CGRAF     39
     1    LRID     ,LRPOEQ   /                                          CGRAF     40
     2    20,5*20  ,20,5*21  ,20,5*22  ,20,5*23  ,20,5*24  ,20,5*25  ,  CGRAF     41
     3    20,5*26  ,20,5*27  /                                          CGRAF     42
C                                                                       CGRAF     43
C    THIS DATA STATEMENT CONTAINS THE SIX INTEGER ARRAYS IDENTIFYING THE CGRAF    44
C    TEXT ENTITIES DISPLAYING BLINKING ASTERISKS IN THE LOWER LEFT HAND CGRAF     45
C    CORNER OF THE SCREEN                                               CGRAF     46
      DATA NCUPS1  ,NCUPS2   ,NCAFU2   ,NCAFL2   ,NCAFU3   ,NCDWN1   ,  CGRAF     47
     1    NCDWN2   /                                                    CGRAF     48
     2    30,5*30  ,30,5*31  ,30,5*32  ,30,5*33  ,30,5*34  ,30,5*35  ,  CGRAF     49
     3    30,5*36  /                                                    CGRAF     50
C                                                                       CGRAF     51
C    THIS DATA STATEMENT CONTAINS THE SIX INTEGER ARRAYS IDENTIFYING THE CGRAF    52
C    TEXT ENTITIES DISPLAYING NONBLINKING ASTERISKS IN THE LOWER LEFT   CGRAF     53
C    HAND CORNER OF THE SCREEN                                          CGRAF     54
```

```
      DATA NPUPS1  ,NPUPS2    ,NPAFU1    ,NPAFU2    ,NPAFU3    ,NPAFL1       CGRAF    55
     1    ,NPAFL2   ,NPDWN1    ,NPDWN2    ,NUPS      ,NAF1      ,NAF2     /   CGRAF    56
     2     31,32,37,34,37,31   ,31,32,33,37,35,34   ,32,32,33,34,35,33         CGRAF    57
     3    ,6*32                ,6*33                ,32,32,33,34,35,36         CGRAF    58
     4    ,6*34                ,6*35                ,6*36                      CGRAF    59
     5    ,31,32,4*0           ,0,32,33,3*0         ,0,32,0,34,2*0        /    CGRAF    60
C                                                                              CGRAF    61
C    TEXT ENTITIES DISPLAYING CHARACTERS WHICH IDENTIFY THE ABSCISSA AND       CGRAF    62
C    ORDINATE AXES OF THE GRAPHICAL OUTPUT                                     CGRAF    63
      DATA NX1B     ,NX2B      ,NYB       ,NMB       ,NM01B     ,NM02B     ,   CGRAF    64
     1     NDU1B    ,NDU2B     ,NDD0B     ,NPOB      ,NP1B      ,NPKTAB    /   CGRAF    65
     2     40,41,42,43,44,45   ,40,3*47,44,45       ,40,41,4*47            ,   CGRAF    66
     3     40,41,4*48          ,40,46,42,3*46       ,40,41,4*49            ,   CGRAF    67
     4     40,2*46,43,2*46     ,40,3*46,44,46       ,40,3*47,44,47         ,   CGRAF    68
     5     40,4*48,45          ,40,4*49,45          ,49,41,42,43,44,45     /   CGRAF    69
      DATA NMXB     ,NUPB      ,NDUDXB    ,NAF3B     ,NDWNB     ,NKTAB     /   CGRAF    70
     1     40,0,42,3*0         ,40,41,4*0           ,40,2*0,43,2*0         ,   CGRAF    71
     2     40,3*0,44,0         ,40,4*0,45           ,0,41,42,43,44,45      /   CGRAF    72
C                                                                              CGRAF    73
C    THIS DATA STATEMENT CONTAINS A SIX INTEGER ARRAYIDENTIFYING A POLY-       CGRAF    74
C    LINE ENTITYWHICH DISPLAYS THE AIRFOIL SHAPE IN THE LOWER LEFT HAND        CGRAF    75
C    CORNER OF THE SCREEN                                                      CGRAF    76
      DATA NAIRFL/6*18        /                                                CGRAF    77
      DATA LRSUPR   ,LRSUB     ,LRFLOW    /2*19,4*1 ,2*19,4*2 ,2*19,4*0 /      CGRAF    78
      DATA NLGRNG   ,NPARAB    /5*39,37   ,5*39,38   /                         CGRAF    79
      DATA LRNOGC   /6*39      /                                               CGRAF    80
C                                                                              CGRAF    81
C    THIS DATA STATEMENT CONTAINS THE SIX INTEGER ARRAYS IDENTIFYING THE       CGRAF    82
C    POLYLINE ENTITIES FOR GRAPHICAL OUTPUT                                    CGRAF    83
      DATA NXY1     ,NXY2      /6*60      ,6*61      /                         CGRAF    84
      DATA NALL/6*0/                                                           CGRAF    85
      END                                                                      CGRAF    86
```

```
      OVERLAY(OVFILE,0,0)                                               LIEN       2
      PROGRAM LIEN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)               LIEN       3
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      COMMON/NCON/  ICON                                                NCON       2
      COMMON/OUTCOM/                                                    OUTCOM     2
     1    X1(160)  ,Y1(160)   ,Y2(160)   ,NN1       ,NN2                OUTCOM     3
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY     2
      COMMON/ICNTRL/J         ,ICRIT(2) ,LL(2)     ,IGO(2)              ICNTRL     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LROIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(5)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NCHARS/NNEQ      ,NAEQ     ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2)   NCHARS     2
     1   ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)    NCHARS     3
     2   ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)   ,NMACHQ(2),NALPHA(2)   NCHARS     4
     3   ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI     ,FMTF                  NCHARS     5
      COMMON C     ,CK       ,RS       ,FM       ,ALPHA                 COMMON     2
      COMMON/BCOM/ XO         ,DVOO     ,L                              BCOM       2
      COMMON/ECOM/ASTAG(26),XSTAG(11),YSTAG(11),XAF(50),YAF(50,2)       ECOM       2
      COMMON/AINPUT/AIN(24)   ,NNI(7)   ,HI(6)                          AINPUT     2
      COMMON/YUVSAV/NNN(3)    ,YUV(126)                                 YUVSAV     2
      COMMON/PTARFL/XX(40,2)  ,YY(40,2) ,AM(40,2) ,CA       ,SA         PTARFL     2
      COMMON/COMPRS/XP(160,2),PP(160,2),NP(2)                           COMPRS     2
      COMMON/RBUBCM/RBUB      ,UBINIT   ,IRBUB                          RBUBCM     2
C                                                                       LIEN      22
C   INITIATE PROGRAM EXECUTION                                          LIEN      23
      CALL GPEXE(4LNPUT)                                                LIEN      24
      END                                                               LIEN      25
```

```
      OVERLAY(1,0)                                                        NPUT        2
      PROGRAM NPUT                                                        NPUT        3
C                                                                         NPUT        4
C   THIS PROGRAM READS INPUT                                              NPUT        5
C                                                                         NPUT        6
      COMMON/NCON/  ICON                                                  NCON        2
      COMMON/AINPUT/AIN(24)   ,NNI(7)   ,HI(6)                            AINPUT      2
      COMMON/YUVSAV/NNN(3)    ,YUV(126)                                   YUVSAV      2
      COMMON/PTARFL/XX(40,2) ,YY(40,2) ,AM(40,2) ,CA      ,SA             PTARFL      2
      COMMON C      ,CK       ,RS       ,FM       ,ALPHA                  COMMON      2
      DIMENSION LABEL(7),NP(2)                                            NPUT       12
C                                                                         NPUT       13
C   READ THE CONSOLE NUMBER AND THE LABEL CARD                            NPUT       14
      READ(5,90) ICON,(LABEL(I),I=1,7)                                    NPUT       15
C                                                                         NPUT       16
C   READ IN FLOW CONDITIONS                                               NPUT       17
      READ(5,100) FM,TC,ALPHA                                             NPUT       18
C                                                                         NPUT       19
C   READ IN FLOW SOLUTION PARAMETERS                                      NPUT       20
      READ(5,210) (AIN(I),I=1,7)                                          NPUT       21
      READ(5,100) (AIN(I),I=8,15)                                         NPUT       22
      READ(5,100) (AIN(I),I=16,18)                                        NPUT       23
C                                                                         NPUT       24
C   READ IN THE NUMBER OF STRIPS AND THE INTEGRATION STEP SIZE            NPUT       25
      READ(5,110) (NNI(I),I=1,6),(HI(I),I=1,6)                            NPUT       26
      WRITE(6,400)                                                        NPUT       27
      WRITE(6,330) (LABEL(I),I=1,7)                                       NPUT       28
      READ(5,140) NP(1),NP(2)                                             NPUT       29
      DO 20 J=1,2                                                         NPUT       30
      NN = NP(J)                                                          NPUT       31
   20 READ(5,150) (XX(I,J),YY(I,J),AM(I,J),I=1,NN)                        NPUT       32
      NN1 = NP(1)                                                         NPUT       33
      IF(NP(2).LT.NP(1)) NN1 = NP(2)                                      NPUT       34
      WRITE(6,240)                                                        NPUT       35
      DO 30 I=1,NN1                                                       NPUT       36
   30 WRITE(6,250)  XX(I,1),YY(I,1),AM(I,1),XX(I,2),YY(I,2),AM(I,2)       NPUT       37
      NN1 = NN1+1                                                         NPUT       38
      J = 1                                                               NPUT       39
      IF(NP(2).GT.NP(1)) J=2                                              NPUT       40
      IF(NP(2).EQ.NP(1)) GO TO 80                                         NPUT       41
      NN2 = NP(J)                                                         NPUT       42
      GO TO (40,60),J                                                     NPUT       43
   40 DO 50 I=NN1,NN2                                                     NPUT       44
   50 WRITE(6,260) XX(I,1),YY(I,1),AM(I,1)                                NPUT       45
      GO TO 80                                                            NPUT       46
   60 DO 70 I=NN1,NN2                                                     NPUT       47
   70 WRITE(6,270) XX(I,2),YY(I,2),AM(I,2)                                NPUT       48
   80 CA = COS(ALPHA/57.2957795)                                          NPUT       49
      SA = SIN(ALPHA/57.2957795)                                          NPUT       50
      CALL AETSKC(4LSTUP)                                                 NPUT       51
   90 FORMAT(I2,8X,7A10)                                                  NPUT       52
  100 FORMAT(8F10.6)                                                      NPUT       53
  110 FORMAT(6I1,4X,7F10.6)                                               NPUT       54
  140 FORMAT(2I2)                                                         NPUT       55
  210 FORMAT(10X,7F10.6)                                                  NPUT       56
  150 FORMAT(3F20.15)                                                     NPUT       57
  240 FORMAT(17X,8HX(UPPER),12X,8HY(UPPER),9X,12HDY/DX(UPPER),11X,        NPUT       58
     1  8HX(LOWER),12X,8HY(LOWER),9X,12HDY/DX(LOWER))                     NPUT       59
  250 FORMAT(7X,6F20.12)                                                  NPUT       60
  260 FORMAT(7X,3F20.12)                                                  NPUT       61
  270 FORMAT(67X,3F20.12)                                                 NPUT       62
  330 FORMAT(4(/)32X,7A10,4(/))                                           NPUT       63
  400 FORMAT(1H1,4(/)7X,12(10H**********))                                NPUT       64
      END                                                                 NPUT       65
```

```
      OVERLAY(2,0)                                                      STUP       2
      PROGRAM STUP                                                      STUP       3
C                                                                       STUP       4
C   THIS PROGRAM SETS UP THE LIGHT REGISTERS, THE LIGHT BUTTONS,AND     STUP       5
C   MANY TEXT ENTITIES                                                  STUP       6
      COMMON/NCON/  ICON                                                NCON       2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP13(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NCHARS/NNEQ     ,NAEQ     ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2)    NCHARS     2
     1   ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)    NCHARS     3
     2   ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)   ,NMACHQ(2),NALPHA(2)   NCHARS     4
     3   ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI     ,FMTF                  NCHARS     5
      COMMON/AINPUT/AIN(24)  ,NNI(7)   ,HI(6)                           AINPUT     2
      COMMON/YUVSAV/NNN(3)   ,YUV(126)                                  YUVSAV     2
      COMMON C      ,CK       ,RS       ,FM       ,ALPHA                COMMON     2
      DIMENSION     IXLR(2)  ,IYLR(4)                     ,XLR(2)   ,   STUP      17
     1    YLR(4)   ,XPRCD(6) ,YPRCD(3) ,XGRID(2) ,YGRID(4) ,ALIM1(4) ,   STUP      18
     2    USER1(4) ,BASE(4)  ,X(60)    ,Y(60)    ,NSNCPB(2),NOGOB(3) ,   STUP      19
     3    IVEL1(2) ,IVEL2(2)                                            STUP      20
      DATA NSNCPB,NOGOB,ICOMP,NSTOP,NSUB,NSUPR,LRSTAR,IPRCD,IVEL1,IVEL2/ STUP      21
     1    10HSONIC POIN,10HT REACHED ,10HINTEGRATIO,10HN INCOMPLE,      STUP      22
     2    2HTE         ,7HCOMPUTE    ,4HSTOP       ,8HSUBSONIC    ,     STUP      23
     3    10HSUPERSONIC,1H*          ,7HPROCEED    ,10HLAGRANGIAN,      STUP      24
     4    10H FUNCTION ,10HPARABOLIC ,10HFUNCTION  /                    STUP      25
      DATA     NX,NY,NM,NMO,NDUDX,NDDQ,NPO,NP1                 /        STUP      26
     1    1HX,1HY,1HM,2HMO,4HDUDX,3HDDQ,2HPO,2HP1              /        STUP      27
      DATA IXLR,IYLR,XLR,YLR,XPRCD,YPRCD                       /        STUP      28
     1      11 ,36 ,-53 ,-49 ,-45 ,-41 ,                                STUP      29
     2      10.,35.,-56.,-52.,-48.,-44.,                                STUP      30
     3      -56.,-50.,-39.,-37.,-27.,-2.,-56.,-50.,-43.        /        STUP      31
      DATA      XGRID,YGRID,ALIM1,USER1,BASE                   /        STUP      32
     1    -54.,1.,-42.,-16.,12.,36.,3*(-57.,-57.,57.,57.) /             STUP      33
      DATA IXCM,IYCM/35,-56/                                            STUP      34
C                                                                       STUP      35
C   INITIALIZE SUBSCREEN AREA IDENTIFIER                                STUP      36
      IDSCAL = 1                                                        STUP      37
C                                                                       STUP      38
C   CREATE POINTS ON AIRFOIL FOR THE POLYLINE ENTITY NAIRFL             STUP      39
      X(1) = -12.                                                       STUP      40
      CALL ARFL(1.0,A,YY,B,C,2)                                         STUP      41
      Y(1) = -50.-YY*25.                                                STUP      42
      XX = 0.95                                                         STUP      43
      DO 5 I=2,29                                                       STUP      44
      CALL ARFL(XX,A,YY,B,C,2)                                          STUP      45
      X(I) = -37.+XX*25.                                                STUP      46
      Y(I) = -50.-YY*25.                                                STUP      47
      DX = -.05                                                         STUP      48
      IF(I.GE.19) DX = -.01                                             STUP      49
      IF(I.GE.28) DX = -.005                                            STUP      50
    5 XX = XX+DX                                                        STUP      51
      X(30) = -37.                                                      STUP      52
      Y(30) = -50.                                                      STUP      53
      XX = .005                                                         STUP      54
```

```
      DO 6 I=31,58
      CALL ARFL(XX,A,YY,B,C,1)
      X(I) = -37.+XX*25.
      Y(I) = -50.+YY*25.
      DX = .005
      IF(I.GE.32) DX = .01
      IF(I.GE.41) DX = .05
    6 XX = XX+DX
      X(59) = -12.
      Y(59) = Y(1)
C
C   INITIALIZE GRAPHIC PAC FACILITIES
      CALL INTGP(6LDATAFL,3,510,1)
C
C   INITIALIZE THE GRAPHICS CONSOLE NUMBER ICON
      CALL INCON(ICON)
C
C   DEFINE SUBSCREEN AREA 1 TO DEFINE THE ENTIRE SCREEN AREA
      CALL SCORS(BASE)
      CALL SSCAL(1,ALIM1,USER1)
      CALL PLYLN(NAIRFL,1,X(1),Y(1),58)
      CALL ASCAL(1)
      CALL GENDF(NAIRFL,0)
C
C   CREATE POINT ENTITIES IN THE LOWER RIGHT HAND CORNER OF THE SCREEN
      CALL POINT(XLR(1),YLR(1),PTLR11)
      CALL POINT(XLR(1),YLR(2),PTLR12)
      CALL POINT(XLR(1),YLR(3),PTLR13)
      CALL POINT(XLR(1),YLR(4),PTLR14)
      CALL POINT(XLR(2),YLR(1),PTLR21)
      CALL POINT(XLR(2),YLR(2),PTLR22)
      CALL POINT(XLR(2),YLR(3),PTLR23)
      CALL POINT(XLR(2),YLR(4),PTLR24)
C
C   CREATE POINT ENTITIES IN THE LOWER LEFT HAND CORNER OF THE SCREEN
      CALL POINT(XPRCD(2),YPRCD(2),ISTR22)
      CALL POINT(XPRCD(3),YPRCD(2),ISTR32)
      CALL POINT(XPRCD(4),YPRCD(3),ISTR43)
      CALL POINT(XPRCD(5),YPRCD(3),ISTR53)
      CALL POINT(XPRCD(4),YPRCD(1),ISTR41)
      CALL POINT(XPRCD(5),YPRCD(1),ISTR51)
      CALL POINT(XPRCD(5),YPRCD(2),ISTR52)
      CALL POINT(XPRCD(6),YPRCD(2),ISTR62)
C
C   CREATE POINT ENTITIES IN THE UPPER AREA OF THE SCREEN
      CALL POINT(XGRID(2),YGRID(1),PGRD21)
      CALL POINT(XGRID(1),YGRID(2),PGRD12)
      CALL POINT(XGRID(1),YGRID(3),PGRD13)
      CALL POINT(XGRID(2),YGRID(3),PGRD23)
      CALL POINT(XGRID(1),YGRID(4),PGRD14)
C
C   CREATE LIGHT REGISTERS FOR THE LOWER RIGHT HAND CORNER OF THE SCREEN
C   WHICH CAN BE USED TO CHANGE INPUT VARIABLES
      CALL LITRG(IXLR(2),IYLR(2),NNEQ,3,21,FMTI)
      CALL LITRG(IXLR(2),IYLR(2),NAEQ,3,22,FMTI)
      CALL LITRG(IXLR(2),IYLR(2),NNEQ,3,23,FMTI)
      CALL LITRG(IXLR(2),IYLR(2),NNEQ,3,24,FMTI)
      CALL LITRG(IXLR(2),IYLR(2),NNEQ,3,25,FMTI)
      CALL LITRG(IXLR(2),IYLR(2),NNEQ,3,26,FMTI)
      CALL LITRG(IXLR(2),IYLR(4),NXOOEQ,4,1,FMTF)
      CALL LITRG(IXLR(2),IYLR(4),NOVIEQ,8,2,FMTF)
      CALL LITRG(IXLR(2),IYLR(3),NXSEQ,3,3,FMTF)
      CALL LITRG(IXLR(2),IYLR(4),NXAEQ,3,4,FMTF)
      CALL LITRG(IXLR(2),IYLR(3),NCYDEQ,4,5,FMTF)
      CALL LITRG(IXLR(2),IYLR(4),NXAEQ,3,6,FMTF)
      CALL LITRG(IXLR(2),IYLR(3),NCYDEQ,4,7,FMTF)
      CALL LITRG(IXLR(2),IYLR(4),NSLEQ,8,8,FMTF)
      CALL LITRG(IXLR(2),IYLR(4),NMACHQ,9,11,FMTF)
      CALL LITRG(IXLR(2),IYLR(3),NALPHA,6,12,FMTF)
      CALL LITRG(IXLR(2),IYLR(2),NYIUEQ,8,13,FMTF)
```

```
      CALL LITRG(IXLR(2),IYLR(1),NYILEQ,8,14,FMTF)                      STUP   125
C                                                                       STUP   126
C   CREATE TEXT ENTITIES FOR THE LOWER RIGHT HAND CORNER OF THE SCREEN  STUP   127
C   WHICH CAN BE USED IN CONJUNCTION WITH THE LIGHT REGISTERS TO CHANGE STUP   128
C   INPUT VARIABLES                                                     STUP   129
      CALL ENSHFT(NNEQ,3,NNI(1),FMTI)                                   STUP   130
      CALL TEXT(LRNN1,0,PTLR22,NNEQ,4,0,3,4RCVLI)                       STUP   131
      CALL ENSHFT(NAEQ,3,NNI(2),FMTI)                                   STUP   132
      CALL TEXT(LRNA2,0,PTLR23,NAEQ,4,0,3,4RCVLI)                       STUP   133
      CALL ENSHFT(NNEQ,3,NNI(3),FMTI)                                   STUP   134
      CALL TEXT(LRNN3,0,PTLR22,NNEQ,4,0,3,4RCVLI)                       STUP   135
      CALL ENSHFT(NNEQ,3,NNI(4),FMTI)                                   STUP   136
      CALL TEXT(LRNN4,0,PTLR22,NNEQ,4,0,3,4RCVLI)                       STUP   137
      CALL ENSHFT(NNEQ,3,NNI(5),FMTI)                                   STUP   138
      CALL TEXT(LRNN5,0,PTLR22,NNEQ,4,0,3,4RCVLI)                       STUP   139
      CALL ENSHFT(NNEQ,3,NNI(6),FMTI)                                   STUP   140
      CALL TEXT(LRNN6,0,PTLR22,NNEQ,4,0,3,4RCVLI)                       STUP   141
      CALL ENSHFT(NDVIEQ,8,AIN(1),FMTF)                                 STUP   142
      CALL TEXT(LRDIEQ,0,PTLR24,NDVIEQ,18,0,3,4RCVLR)                   STUP   143
      CALL ENSHFT(NXSEQ,3,AIN(2),FMTF)                                  STUP   144
      CALL TEXT(LRXSEQ,0,PTLR23,NXSEQ,13,0,3,4RCVLR)                    STUP   145
      CALL ENSHFT(NXOOEQ,4,AIN(8),FMTF)                                 STUP   146
      CALL TEXT(LRXOOQ,0,PTLR24,NXOOEQ,14,0,3,4RCVLR)                   STUP   147
      CALL ENSHFT(NXAEQ,3,AIN(3),FMTF)                                  STUP   148
      CALL TEXT(LRXAUP,0,PTLR24,NXAEQ,13,0,3,4RCVLR)                    STUP   149
      CALL TEXT(LRCYDU,0,PTLR23,NCYDEQ,14,0,3,4RCVLR)                   STUP   151
      CALL ENSHFT(NXAEQ,3,AIN(5),FMTF)                                  STUP   152
      CALL TEXT(LRXALW,0,PTLR24,NXAEQ,13,0,3,4RCVLR)                    STUP   153
      CALL ENSHFT(NCYDEQ,4,AIN(6),FMTF)                                 STUP   154
      CALL TEXT(LRCYDL,0,PTLR23,NCYDEQ,14,0,3,4RCVLR)                   STUP   155
      CALL ENSHFT(NSLEQ,8,AIN(7),FMTF)                                  STUP   156
      CALL TEXT(LRSLEQ,0,PTLR24,NSLEQ,18,0,3,4RCVLR)                    STUP   157
      CALL ENSHFT(NMACHQ,9,FM,FMTF)                                     STUP   158
      CALL TEXT(LRMACH,0,PTLR24,NMACHQ,19,0,3,4RCVLR)                   STUP   159
      CALL ENSHFT(NALPHA,6,ALPHA,FMTF)                                  STUP   160
      CALL TEXT(LRALFA,0,PTLR23,NALPHA,16,0,3,4RCVLR)                   STUP   161
      CALL ENSHFT(NYIUEQ,8,AIN(11),FMTF)                                STUP   162
      CALL TEXT(LRYIU,0,PTLR22,NYIUEQ,18,0,3,4RCVLR)                    STUP   163
      CALL ENSHFT(NYILEQ,8,AIN(12),FMTF)                                STUP   164
      CALL TEXT(LRYIL,0,PTLR21,NYILEQ,18,0,3,4RCVLR)                    STUP   165
C                                                                       STUP   166
C   CREATE TEXT ENTITIES FOR THE LOWER RIGHT HAND CORNER OF THE SCREEN  STUP   167
C   WHICH DISPLAY OUTPUT INFORMATION                                    STUP   168
      CALL TEXT(LRDEEQ,1,PTLR14,NDEEQ,13)                               STUP   169
      CALL TEXT(LRYSOQ,1,PTLR13,NYSOEQ,14)                              STUP   170
      CALL TEXT(LRYSEQ,1,PTLR14,NYSEQ,13)                               STUP   171
      CALL TEXT(LRDOEQ,1,PTLR13,NDVOEQ,18)                              STUP   172
      CALL TEXT(LRRUEQ,1,PTLR14,NRUEQ,16)                               STUP   173
      CALL TEXT(LRUBEQ,1,PTLR13,NUBEQ,13)                               STUP   174
      CALL TEXT(LRID   ,1,PTLR13,NID,20)                                STUP   175
      CALL TEXT(LRPOEQ,1,PTLR12,NPOEQ,13)                               STUP   176
C                                                                       STUP   177
C   CREATE TEXT ENTITIES IN THE UPPER AREA OF THE SCREENWHICHLABEL THE  STUP   178
C   GRAPHIC DISPLAY                                                     STUP   179
      CALL TEXT(NX1B,1,PGRD21,NX,1)                                     STUP   180
      CALL TEXT(NX2B,1,PGRD23,NX,1)                                     STUP   181
      CALL TEXT(NYB  ,1,PGRD14,NY,1)                                    STUP   182
      CALL TEXT(NMB  ,1,PGRD23,NM,1)                                    STUP   183
      CALL TEXT(NMO1B,1,PGRD13,NMO,2)                                   STUP   184
      CALL TEXT(NMO2B,1,PGRD12,NMO,2)                                   STUP   185
      CALL TEXT(NDU1B,1,PGRD13,NDUDX,4)                                 STUP   186
      CALL TEXT(NDU2B,1,PGRD12,NDUDX,4)                                 STUP   187
      CALL TEXT(NDDQB,1,PGRD14,NDDQ,3)                                  STUP   188
      CALL TEXT( NPOB,1,PGRD12,NPO,2)                                   STUP   189
      CALL TEXT(NPKTAB,1,PGRD13,NPO,2)                                  STUP   190
      CALL TEXT( NP1B,1,PGRD14,NP1,2)                                   STUP   191
C                                                                       STUP   192
C   CREATE TEXT ENTITIES IN THE LOWER LEFT HAND CORNER OF THE SCREEN    STUP   193
C   WHICHDISPLAY ASTERISKS INDICATING NEXT PROGRAM STEP                 STUP   194
```

```
      CALL TEXT(NPUPS1,0,ISTR22,LRSTAR,1,0,3,4RSTRT)
      CALL TEXT(NPUPS2,0,ISTR32,LRSTAR,1,0,3,4RPRP2)
      CALL TEXT(NPAFU1,0,ISTR43,LRSTAR,1,0,3,4RAFU1)
      CALL TEXT(NPAFU2,0,ISTR43,LRSTAR,1,0,3,4RPRP3)
      CALL TEXT(NPAFU3,0,ISTR53,LRSTAR,1,0,3,4RPRP5)
      CALL TEXT(NPAFL1,0,ISTR41,LRSTAR,1,0,3,4RAFL1)
      CALL TEXT(NPAFL2,0,ISTR51,LRSTAR,1,0,3,4RPRP4)
      CALL TEXT(NPDWN1,0,ISTR62,LRSTAR,1,0,3,4RPRP6)
      CALL TEXT(NPDWN2,0,ISTR52,LRSTAR,1,0,3,4RDWN2)
C
C   CREATE TEXT ENTITIES IN THE LOWER LEFT HAND CORNER OF THE SCREEN
C   WHICH DISPLAY BLINKING ASTERISKS INDICATING THE CURRENT PROGRAM STEP
      CALL TEXT(NCUPS1,1,ISTR22,LRSTAR,1,1,3)
      CALL TEXT(NCUPS2,1,ISTR32,LRSTAR,1,1,3)
      CALL TEXT(NCAFU2,1,ISTR43,LRSTAR,1,1,3)
      CALL TEXT(NCAFL2,1,ISTR51,LRSTAR,1,1,3)
      CALL TEXT(NCAFU3,1,ISTR53,LRSTAR,1,1,3)
      CALL TEXT(NCDWN1,1,ISTR62,LRSTAR,1,1,3)
      CALL TEXT(NCDWN2,1,ISTR52,LRSTAR,1,1,3)
      CALL POINT(10.,0.,PTGRF2)
      CALL TEXT(LRNOGO,1,PTGRF2,NOGOB,22)
      CALL TEXT(LRSUB,0,PTLR24,NSUB,8,0,3,4RPRP4)
      CALL TEXT(LRSUPR,0,PTLR23,NSUPR,10,0,3,4RPRP3)
      CALL TEXT(NLGRNG,0,PTLR22,IVEL1,19,0,3,4RCHGV)
      CALL TEXT(NPARAB,0,PTLR22,IVEL2,18,0,3,4RCHGV)
      CALL LITBN(IXLR(1),IYLR(1),IPRCD,7,1,4RPRP1)
C
C   CREATE LIGHT BUTTONS FOR COMPUTATION OF THE CURRENT PROGRAM STEP
      CALL LITBN(IXCM,IYCM,ICOMP,7,2,4RUPS1)
      CALL LITBN(IXCM,IYCM,ICOMP,7,3,4RUPS2)
      CALL LITBN(IXCM,IYCM,ICOMP,7,4,4RAFU2)
      CALL LITBN(IXCM,IYCM,ICOMP,7,5,4RAFL2)
      CALL LITBN(IXCM,IYCM,ICOMP,7,6,4RAFU3)
      CALL LITBN(IXCM,IYCM,ICOMP,7,7,4RDWN1)
      CALL LITBN(-56,-56,NSTOP,4,20,4RSTOP)
C
C   ENABLE ALL TEXT AND POLYLINE ENTITIES
      CALL ENGDS(1,NALL)
      CALL AETSKC(4LSTRT)
      END
```

```
      OVERLAY(3,0)                                                      STRT       2
      PROGRAM STRT                                                      STRT       3
C                                                                       STRT       4
C    THIS PROGRAM DISPLAYS THE FLOW VARIABLES                           STRT       5
C                                                                       STRT       6
      COMMON/INPUT/                                                     INPUT      2
     1     LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)  INPUT      3
     2    ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)  INPUT      4
     3    ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)  INPUT      5
     4    ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)   INPUT      6
     5    ,NLGRNG(6),NPARAB(6)                                          INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1    ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2    ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)  NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)  NPRCD      3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)    NPRCD      4
     3    ,NAF1(6)  ,NAF2(6)                                            NPRCD      5
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1    ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)   NAXES      3
     2    ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)   NAXES      4
     3    ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)  NAXES      5
      COMMON C      ,CK       ,RS       ,FM       ,ALPHA                COMMON     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      DIMENSION LBID(2)                                                 STRT      13
      DATA LBID/1,20/                                                   STRT      14
      CALL ASCAL(1)                                                     STRT      15
      CALL ERASE(NALL)                                                  STRT      16
      CALL ERASG(IDSCAL)                                                STRT      17
      CALL ERASG(IDSCAL-1)                                              STRT      18
      CALL GENDF(NAIRFL,0)                                              STRT      19
      CALL ENLB(2,LBID)                                                 STRT      20
      CALL GENDF(LRSTRT,0)                                              STRT      21
C                                                                       STRT      22
C    WAIT FOR AN ATTENTION SOURCE                                       STRT      23
      CALL WAITE(DUM,0,DUM,DUM)                                         STRT      24
      END                                                               STRT      25
```

```
      OVERLAY(4,0)                                                      PRP1       2
      PROGRAM PRP1                                                      PRP1       3
C                                                                       PRP1       4
C   THIS PROGRAM DISPLAYS ITEMS NEEDED FOR PROGRAM UPS1                 PRP1       5
C                                                                       PRP1       6
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON C     ,CK       ,RS       ,FM       ,ALPHA                 COMMON     2
      COMMON/AINPUT/AIN(24)  ,NNI(7)   ,HI(6)                           AINPUT     2
      COMMON/YUVSAV/NNN(3)   ,YUV(126)                                  YUVSAV     2
      COMMON/PTARFL/XX(40,2) ,YY(40,2) ,AM(40,2) ,CA       ,SA          PTARFL     2
      DIMENSION LBID(2)                                                 PRP1      15
      DATA LBID/2,20/                                                   PRP1      16
      CALL ASCAL(1)                                                     PRP1      17
      CALL ENLB(2,LBID)                                                 PRP1      18
      CALL ERASE(LRSTRT)                                                PRP1      19
      CALL GENDF(NCUPS1,0)                                              PRP1      20
      CALL GENDF(NPUPS2,0)                                              PRP1      21
      CALL GENDF(LRUPS,0)                                               PRP1      22
      CALL GENDF(NUPB,0)                                                PRP1      23
      WRITE(6,100) FM,ALPHA                                             PRP1      24
  100 FORMAT(1H1///20X,9HMACH NO.=,F10.6,16H          ALPHA=,F10.6)     PRP1      25
      CALL WAITE(DUM,0,DUM,DUM)                                         PRP1      26
      END                                                               PRP1      27
```

```
      OVERLAY(5,0)                                                      UPS1       2
      PROGRAM UPS1                                                      UPS1       3
C                                                                       UPS1       4
C   THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE UPSTRM                 UPS1       5
C                                                                       UPS1       6
      COMMON/OUTCOM/                                                    OUTCOM     2
     1    X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2               OUTCOM     3
      COMMON/COMNXY/NXY1(6)    ,NXY2(6)                                 COMNXY     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)  INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYOL(6)  INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)  INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)   INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)  NOUT       4
      COMMON/NCHARS/NNEQ       ,NAEQ       ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2) NCHARS    2
     1   ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)   NCHARS     3
     2   ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)    ,NMACHQ(2),NALPHA(2) NCHARS     4
     3   ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI      ,FMTF                 NCHARS     5
      COMMON C      ,CK         ,RS         ,FM         ,ALPHA          COMMON     2
      COMMON/BCOM/ XO           ,DVOO       ,L                          BCOM       2
      COMMON/ECOM/ASTAG(26),XSTAG(11),YSTAG(11),XAF(50),YAF(50,2)       ECOM       2
      COMMON/AINPUT/AIN(24)    ,NNI(7)     ,HI(6)                       AINPUT     2
      COMMON/YUVSAV/NNN(3)     ,YUV(156)                                YUVSAV     2
      COMMON/PTARFL/XX(40,2) ,YY(40,2) ,AM(40,2) ,CA        ,SA         PTARFL     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      DIMENSION X2TEMP(10),Y2TEMP(10),X2(10)                            UPS1      19
      C = 1.+5./FM**2                                                   UPS1      20
      CK = 1./(1.4*FM*FM)                                               UPS1      21
      CA = COS(ALPHA/57.2957795)                                        UPS1      22
      SA = SIN(ALPHA/57.2957795)                                        UPS1      23
      CALL ASCAL(1)                                                     UPS1      24
      CALL ERASG(IDSCAL)                                                UPS1      25
      CALL ERASG(IDSCAL-1)                                              UPS1      26
      CALL IOUPSTM                                                      UPS1      27
      CALL ERASE(LRDEEQ)                                                UPS1      28
      CALL ERASE(LRYSOQ)                                                UPS1      29
C                                                                       UPS1      30
C   DISPLAY THE OUTPUT VARIABLES DE AND YSO                             UPS1      31
      CALL ENSHFT(NDEEQ,3,AIN(20),FMTF)                                 UPS1      32
      CALL MODFY(LRDEEQ,1,2,NDEEQ)                                      UPS1      33
      CALL GENDF(LRDEEQ,0)                                              UPS1      34
      CALL ENSHFT(NYSOEQ,4,AIN(21),FMTF)                                UPS1      35
      CALL MODFY(LRYSOQ,1,2,NYSOEQ)                                     UPS1      36
      CALL GENDF(LRYSOQ,0)                                              UPS1      37
C                                                                       UPS1      38
C   GRAPHICALLY DISPLAY DATA POINTS IN ARRAY X1-Y1                      UPS1      39
      CALL PLOTT2(0.,1.0,0.0,0.8,0.0,1.0,1)                             UPS1      40
      IF(NN2.GT.1) GO TO 10                                             UPS1      41
C                                                                       UPS1      42
C   GRAPHICALLY DISPLAY DATA POINTS IN ARRAY X2-Y2                      UPS1      43
      NN2 = 4                                                           UPS1      44
      Y2(2) = 1.                                                        UPS1      46
      Y2(3) = 0.                                                        UPS1      47
      Y2(4) = 1.                                                        UPS1      48
      Y2(11) = 0.                                                       UPS1      49
      Y2(12) = 1.                                                       UPS1      50
      Y2(13) = 1.                                                       UPS1      51
      Y2(14) = 0.                                                       UPS1      52
   10 DO 15 I=1,NN2                                                     UPS1      53
      X2TEMP(I) = Y2(I)                                                 UPS1      54
   15 Y2TEMP(I) = Y2(I+10)                                              UPS1      55
      NSWITCH = NN2-1                                                   UPS1      56
      DO 18 I=1,NN2                                                     UPS1      57
      X2(I) = X2TEMP(I+NSWITCH)                                         UPS1      58
      Y2(I) = Y2TEMP(I+NSWITCH)                                         UPS1      59
```

```
   18 NSWITCH = NSWITCH-2
      Y2MAX = Y2(1)
      DO 25 I=2,NN2
      IF(Y2(I)-Y2MAX)25,25,22
   22 Y2MAX = Y2(I)
   25 CONTINUE
      X2MIN = 0.0
      Y2MIN = 0.0
      X2MAX = X2(NN2)
      CALL AREA2(Y2MIN,Y2MAX,X2MIN,X2MAX,2)
      NXY2(5) = 0
      CALL DLETE(NXY2)
      NXY2(5) = 61
      CALL PLYLN(NXY2,1,Y2(1),X2(1),NN2-1)
      CALL GENDF(NXY2,0)
C
C    WAIT FOR AN ATTENTION SOURCE
      CALL WAITE(DUM,0,DUM,DUM)
      END
```

```
      OVERLAY(6,0)                                                      PRP2       2
      PROGRAM PRP2                                                      PRP2       3
C                                                                       PRP2       4
C    THIS PROGRAM DISPLAYS ITEMS NEEDED FOR PROGRAM UPS2                PRP2       5
      COMMON/COMNXY/NXY1(6)    ,NXY2(6)                                 COMNXY     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NM01B(6) ,NM02B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON/ICNTRL/J          ,ICRIT(2) ,LL(2)    ,IGO(2)              ICNTRL     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      DIMENSION LBID(2)                                                 PRP2      13
      DATA LBID/3,20/                                                   PRP2      14
      IGO(1) = 0                                                        PRP2      15
      IGO(2) = 0                                                        PRP2      16
      NXY1(5) = 0                                                       PRP2      17
      NXY1(6) = 0                                                       PRP2      18
      CALL DLETE(NXY1)                                                  PRP2      19
      NXY1(6)=60                                                        PRP2      20
      CALL ASCAL(1)                                                     PRP2      21
      CALL ERASG(IDSCAL)                                                PRP2      22
      CALL ERASG(IDSCAL-1)                                              PRP2      23
      CALL ERASE(NALL)                                                  PRP2      24
      CALL ENLB(2,LBID)                                                 PRP2      25
      CALL GENDF(NAIRFL,0)                                              PRP2      26
      CALL GENDF(NCUPS2,0)                                              PRP2      27
      CALL GENDF(NAF2,0)                                                PRP2      28
      CALL GENDF(LRSTG,0)                                               PRP2      29
      CALL WAITE(DUM,0,DUM,DUM)                                         PRP2      30
      END                                                               PRP2      31
```

```
      OVERLAY(7,0)                                                      UPS2       2
      PROGRAM UPS2                                                      UPS2       3
C                                                                       UPS2       4
C   THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE IOSTGNA                UPS2       5
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1    ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2    ,LRID(6)   ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6) NOUT       4
      COMMON/NCHARS/NNEQ      ,NAEQ      ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2)  NCHARS     2
     1    ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)   NCHARS     3
     2    ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)    ,NMACHQ(2),NALPHA(2) NCHARS     4
     3    ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI      ,FMTF                NCHARS     5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)  NPRCD      3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)    NPRCD      4
     3    ,NAF1(6)   ,NAF2(6)                                           NPRCD      5
      COMMON C       ,CK        ,RS        ,FM        ,ALPHA            COMMON     2
      COMMON/BCOM/ XO          ,DVOO      ,L                            BCOM       2
      COMMON/ECOM/ASTAG(26),XSTAG(11),YSTAG(11),XAF(50),YAF(50,2)       ECOM       2
      COMMON/AINPUT/AIN(24)    ,NNI(7)    ,HI(6)                        AINPUT     2
      COMMON/PTARFL/XX(40,2)  ,YY(40,2)  ,AM(40,2)  ,CA        ,SA      PTARFL     2
      DIMENSION XINF(2),YINF(2),XSTG(11),YSTG(11),XLOWER(50),YLOWER(50), UPS2     14
     1XUPPER(50),YUPPER(50)                                             UPS2      15
      DIMENSION NXYUPR(6),NXYLWR(6),NXYSTG(6),NYOINF(6),NNNALL(6)       UPS2      16
      DATA NXYUPR,NXYLWR,NXYSTG,NYOINF,NNNALL/6*50,50,5*52,50,5*53,     UPS2      17
     1    50,5*54,50,5*0/                                               UPS2      18
      CALL IOSTGNA                                                      UPS2      19
      CALL ASCAL(1)                                                     UPS2      20
      CALL DLETE(NNNALL)                                                UPS2      21
      CALL ERASE(LRDOEQ)                                                UPS2      22
C                                                                       UPS2      23
C   ERASE TEXT ENTITIES PREVIOUSLY DISPLAYED BY THIS PROGRAM            UPS2      24
      CALL ERASE(LRYSEQ)                                                UPS2      25
      YSPYSO =-AIN(21)-AIN(22)                                          UPS2      26
      DE = AIN(20)                                                      UPS2      27
      YLWR = AMIN1(YAF(50,2),YSPYSO)                                    UPS2      28
C                                                                       UPS2      29
C   DETERMINE THE SCALING FACTORS FOR THE SCREEN DISPLAY                UPS2      30
      AMULTX = 114./(XAF(50)   -XSTAG(1))                               UPS2      31
      AMULTY = 100./(-YLWR)                                             UPS2      32
      AMULT = AMULTX                                                    UPS2      33
      IF(AMULTX.GT.AMULTY) AMULT = AMULTY                               UPS2      34
      DE = AIN(20)                                                      UPS2      35
C                                                                       UPS2      36
C   DETERMINE DATA POINTS FOR THE FIRST 3 PER CENT OF THE UPPER AND     UPS2      37
C   LOWER SURFACES OF THE AIRFOIL                                       UPS2      38
      DO 10 I=1,50                                                      UPS2      39
      YUPPER(I) = -40.+(YAF(I,1)-YLWR)*AMULT                            UPS2      40
      IF(YUPPER(I).GT.57.) GO TO 15                                     UPS2      41
      XUPPER(I) = -57.+(XAF(I)   +DE)*AMULT                             UPS2      42
   10 CONTINUE                                                          UPS2      43
      I=51                                                              UPS2      44
   15 CALL PLYLN(NXYUPR,1,XUPPER,YUPPER,I-2)                            UPS2      45
      DO 20 I=1,50                                                      UPS2      46
      XLOWER(I) = -57.+(XAF(I)   +DE)*AMULT                             UPS2      47
      YLOWER(I) = -40.+(YAF(I,2)-YLWR)*AMULT                            UPS2      48
   20 CONTINUE                                                          UPS2      49
      CALL PLYLN(NXYLWR,1,XLOWER,YLOWER,49)                             UPS2      50
C                                                                       UPS2      51
C   DISPLAY THE OUTPUT VARIABLES YS AND DVOOF                           UPS2      52
      CALL ENSHFT(NYSEQ,3,AIN(22),FMTF)                                 UPS2      53
      CALL MODFY(LRYSEQ,1,2,NYSEQ)                                      UPS2      54
      CALL GENDF(LRYSEQ,0)                                              UPS2      55
      CALL ENSHFT(NDVOEQ,8,ASTAG(4),FMTF)                               UPS2      56
      CALL MODFY(LRDOEQ,1,2,NDVOEQ)                                     UPS2      57
      CALL GENDF(LRDOEQ,0)                                              UPS2      58
C                                                                       UPS2      59
C   DETERMINE DATA POINTS FOR THE STAGNATION STREAMLINE                 UPS2      60
      DO 25 I=1,11                                                      UPS2      61
      XSTG(I) = -57.+(XSTAG(I)+DE)*AMULT                                UPS2      62
```

```
      YSTG(I) = -40.+(YSTAG(I)-YLWR)*AMULT                               UPS2   63
   25 CONTINUE                                                           UPS2   64
      CALL PLYLN(NXYSTG,1,XSTG,YSTG,10)                                  UPS2   65
      XINF(1) = -57.                                                     UPS2   66
      XINF(2) = XSTG(11)                                                 UPS2   67
      YINF(1) =  -40.+(YSPYSO-YLWR)*AMULT                                UPS2   68
      YINF(2) = YINF(1)                                                  UPS2   69
      CALL PLYLN(NYOINF,1,XINF,YINF,1)                                   UPS2   70
C                                                                        UPS2   71
C    DISPLAY AIRFOIL NOSE AND STAGNATION STREAMLINE                      UPS2   72
      CALL GENDF(NNNALL,0)                                               UPS2   73
C                                                                        UPS2   74
C    WAIT FOR AN ATTENTION SOURCE                                        UPS2   75
      CALL WAITE(DUM,0,DUM,DUM)                                          UPS2   76
      END                                                                UPS2   77
```

```
      OVERLAY(10,0)                                                     AFU1       2
      PROGRAM AFU1                                                      AFU1       3
C                                                                       AFU1       4
C   THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE IOUPRCT                AFU1       5
C                                                                       AFU1       6
      COMMON/ICNTRL/J          ,ICRIT(2) ,LL(2)     ,IGO(2)             ICNTRL     2
      COMMON/COMNXY/NXY1(6)    ,NXY2(6)                                 COMNXY     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)  INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON C       ,CK       ,RS       ,FM       ,ALPHA               COMMON     2
      COMMON/AINPUT/AIN(24)    ,NNI(7)   ,HI(6)                         AINPUT     2
      COMMON/YUVSAV/NNN(3)     ,YUV(156)                                YUVSAV     2
      COMMON/PTARFL/XX(40,2)   ,YY(40,2) ,AM(40,2) ,CA       ,SA        PTARFL     2
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      NXY1(5) = 0                                                       AFU1      18
      NXY1(6) = 0                                                       AFU1      19
      CALL DLETE(NXY1)                                                  AFU1      20
      NXY1(6) = 60                                                      AFU1      21
      CALL ASCAL(1)                                                     AFU1      22
      CALL ERASG(IDSCAL)                                                AFU1      23
      CALL ERASG(IDSCAL-1)                                              AFU1      24
      CALL ERASE(NALL)                                                  AFU1      25
      CALL ENLB(1,20)                                                   AFU1      26
      CALL GENDF(NAIRFL,0)                                              AFU1      27
      CALL GENDF(NUPS,0)                                                AFU1      28
      J=1                                                               AFU1      29
      LL(J) = 1                                                         AFU1      30
      CALL GENDF(NLGRNG,0)                                              AFU1      31
      CALL GENDF(LRFLOW,0)                                              AFU1      32
      CALL GENDF(NMXB,0)                                                AFU1      33
      CALL IOUPRCT                                                      AFU1      34
      CALL PLOTT1(0.0,0.06,0.0,1.0)                                     AFU1      35
C                                                                       AFU1      36
C   WAIT FOR AN ATTENTION SOURCE                                        AFU1      37
      CALL WAITE(DUN,0,ID,DUM)                                          AFU1      38
      END                                                               AFU1      39
```

```
      OVERLAY(11,0)                                                     PRP3       2
      PROGRAM PRP3                                                      PRP3       3
C                                                                       PRP3       4
C   THIS PROGRAM DISPLAYS ITEMS NEEDED FOR PROGRAM AFU2                 PRP3       5
C                                                                       PRP3       6
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY     2
      COMMON/ICNTRL/J          ,ICRIT(2) ,LL(2)     ,IGO(2)              ICNTRL     2
      COMMON/INPUT/                                                     INPUT      2
     1   LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)    INPUT      3
     2  ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)    INPUT      4
     3  ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)    INPUT      5
     4  ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)     INPUT      6
     5  ,NLGRNG(6),NPARAB(6)                                            INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1  ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)    NOUT       3
     2  ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUR(6) ,LRSUPR(6),LRFLOW(6)    NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1  ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)     NAXES      3
     2  ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)     NAXES      4
     3  ,NDU1B(6) ,NDU2B(6) ,NDDQ3(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)    NAXES      5
      COMMON/NPRCO/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCO      2
     1  ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)    NPRCO      3
     2  ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)      NPRCO      4
     3  ,NAF1(6)  ,NAF2(6)                                              NPRCO      5
      DIMENSION ID(6)                                                   PRP3      13
      DIMENSION LBID(2)                                                 PRP3      14
      DATA LBID/4,20/                                                   PRP3      15
      CALL BPNFO(ITRN,ID)                                               PRP3      16
      J=1                                                               PRP3      17
      ICRIT(J) = ID(3)                                                  PRP3      18
C                                                                       PRP3      19
C   A VALUE OF ICRIT(J)=1 IS ALLOWED FOR THIS VARIABLE, OTHERWISE THE   PRP3      20
C   PROGRAM AWAITS ANOTHER ATTENTION SOURCE                             PRP3      21
      IF(ICRIT(J).EQ.2) CALL WAITE(DUM,0,DUM,DUM)                       PRP3      22
      CALL ASCAL(1)                                                     PRP3      23
      CALL ERASE(NALL)                                                  PRP3      24
      CALL ENLB(2,LBID)                                                 PRP3      25
      CALL GENDF(NAIRFL,0)                                              PRP3      26
      CALL GENDF(NCAFU2,0)                                              PRP3      27
      CALL GENDF(NUPS,0)                                                PRP3      28
      CALL GENDF(LRAFU2,0)                                              PRP3      29
      CALL GENDF(NDUDXB,0)                                              PRP3      30
      CALL WAITE(DUM,0,DUM,DUM)                                         PRP3      31
      END                                                               PRP3      32
```

```
      OVERLAY(12,0)                                                     AFU2       2
      PROGRAM AFU2                                                      AFU2       3
C                                                                       AFU2       4
C   THIS SUBROUTINE DISPLAYS OUTPUT FROM SUBROUTINE IOUPRIN             AFU2       5
      COMMON/OUTCOM/                                                    OUTCOM     2
     1     X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2              OUTCOM     3
      COMMON/ICNTRL/J          ,ICRIT(2)  ,LL(2)     ,IGO(2)            ICNTRL     2
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1    ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2    ,LRID(6)   ,LRPOEQ(6),LRNOGO(6),LRSUB(6)  ,LRSUPR(6),LRFLOW(6) NOUT       4
      COMMON/NCHARS/NNEQ       ,NAEQ      ,NXODEQ(2),NDVIEQ(2),NXSEQ(2) NCHARS     2
     1    ,NYSOEQ(2),NXAEQ(2)  ,NCYDEQ(2),NSLEQ(2)  ,NDEEQ(2)  ,NYSEQ(2)  NCHARS     3
     2    ,NDVOEQ(2),NRUEQ(2)  ,NUBEQ(2)  ,NID(2)    ,NMACHQ(2),NALPHA(2) NCHARS     4
     3    ,NYIUEQ(2),NYILEQ(2),NPOEQ(2)  ,FMTI      ,FMTF               NCHARS     5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)  NPRCD      3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)    NPRCD      4
     3    ,NAF1(6)   ,NAF2(6)                                           NPRCD      5
      COMMON C        ,CK        ,RS        ,FM        ,ALPHA           COMMON     2
      COMMON/AINPUT/AIN(24)    ,NNI(7)    ,HI(6)                        AINPUT     2
      COMMON/YUVSAV/NNN(3)     ,YUV(156)                                YUVSAV     2
      COMMON/PTARFL/XX(40,2)   ,YY(40,2)  ,AM(40,2)  ,CA        ,SA     PTARFL     2
      COMMON/RBUBCM/RBUB       ,UBINIT    ,IRBUB                        RBUBCM     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      CALL ASCAL(1)                                                     AFU2      17
      CALL ERASG(IDSCAL)                                                AFU2      18
      CALL ERASG(IDSCAL-1)                                              AFU2      19
C                                                                       AFU2      20
C   ERASE TEXT ENTITIES PREVIOUSLY DISPLAYED BY THIS PROGRAM            AFU2      21
      CALL ERASE(LRNOGO)                                                AFU2      22
      CALL ERASE(LRRUEQ)                                                AFU2      23
      CALL ERASE(LRID)                                                  AFU2      24
      CALL ERASE(LRUBEQ)                                                AFU2      25
      IICRIT = ICRIT(J)                                                 AFU2      26
      CALL IOUPRIN(IICRIT)                                              AFU2      27
C   DEPENDING ON THE VALUE OF NN2, DISPLAY TEXT ENTITIES RELATING THE   AFU2      28
C   STATUS OF THE INTEGRATION PROCESS                                   AFU2      29
      IF(NN2.NE.1) CALL GENDF(LRNOGO,0)                                 AFU2      30
      IF(NN2.EQ.1) CALL GENDF(NPAFU3,0)                                 AFU2      31
C                                                                       AFU2      32
C   DISPLAY THE OUTPUT VALUES OF RBUB AND UB                            AFU2      33
      CALL ENSHFT(NRUEQ,5,RBUB,FMTF)                                    AFU2      34
      CALL MODFY(LRRUEQ,1,2,NRUEQ)                                      AFU2      35
      CALL GENDF(LRRUEQ,0)                                              AFU2      36
      IF(IRBUB.EQ.0) GO TO 4                                            AFU2      37
      IF(IRBUB.EQ.2) CALL GENDF(LRID,0)                                 AFU2      38
      GO TO 6                                                           AFU2      39
    4 CALL ENSHFT(NUBEQ,3,UBINIT,FMTF)                                  AFU2      40
      CALL MODFY(LRUBEQ,1,2,NUBEQ)                                      AFU2      41
      CALL GENDF(LRUBEQ,0)                                              AFU2      42
    6 CONTINUE                                                          AFU2      43
      DO 10 I=1,NN1                                                     AFU2      44
   10 Y1(I) = Y2(I)                                                     AFU2      45
      CALL PLOTT(0.0,0.04,0.0,20.0)                                     AFU2      46
C                                                                       AFU2      47
C   WAIT FOR AN ATTENTION SOURCE                                        AFU2      48
      CALL WAITE(DUM,0,DUM,DUM)                                         AFU2      49
      END                                                               AFU2      50
```

```
      OVERLAY(13,0)                                                     AFL1       2
      PROGRAM AFL1                                                      AFL1       3
      COMMON/ICNTRL/J          ,ICRIT(2) ,LL(2)    ,IGO(2)              ICNTRL     2
      COMMON/COMNXY/NXY1(6)    ,NXY2(6)                                 COMNXY     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRX00Q(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NM01B(6) ,NM02B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON C      ,CK       ,RS       ,FM       ,ALPHA                COMMON     2
      COMMON/AINPUT/AIN(24)   ,NNI(7)   ,HI(6)                          AINPUT     2
      COMMON/YUVSAV/NNN(3)    ,YUV(156)                                 YUVSAV     2
      COMMON/PTARFL/XX(40,2)  ,YY(40,2) ,AM(40,2) ,CA       ,SA         PTARFL     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      NXY1(5) = 0                                                       AFL1      15
      NXY1(6) = 0                                                       AFL1      16
      CALL DLETE(NXY1)                                                  AFL1      17
      NXY1(6) = 60                                                      AFL1      18
      CALL ASCAL(1)                                                     AFL1      19
      CALL ERASG(IDSCAL)                                                AFL1      20
      CALL ERASG(IDSCAL-1)                                              AFL1      21
      CALL ERASE(NALL)                                                  AFL1      22
      CALL ENLB(1,20)                                                   AFL1      23
      CALL GENDF(NAIRFL,0)                                              AFL1      24
      CALL GENDF(NUPS,0)                                                AFL1      25
      J=2                                                               AFL1      26
      LL(J) = 1                                                         AFL1      27
      CALL GENDF(NLGRNG,0)                                              AFL1      28
      CALL GENDF(LRFLOW,0)                                              AFL1      29
      CALL GENDF(NMXB,0)                                                AFL1      30
      CALL IOLWRCT                                                      AFL1      31
      CALL PLOTT1(0.0,0.06,0.0,1.00)                                    AFL1      32
C                                                                       AFL1      33
C    WAIT FOR AN ATTENTION SOURCE                                       AFL1      34
      CALL WAITE(DUN,0,IC,DUM)                                          AFL1      35
      END                                                               AFL1      36
```

```
      OVERLAY(14,0)                                                     PRP4       2
      PROGRAM PRP4                                                      PRP4       3
C                                                                       PRP4       4
C   THIS PROGRAM DISPLAYS ITEMS NEEDED FOR PROGRAM AFL2                 PRP4       5
C                                                                       PRP4       6
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY     2
      COMMON/ICNTRL/J         ,ICRIT(2) ,LL(2)    ,IGO(2)               ICNTRL     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRJEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(5)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      DIMENSION ID(6)                                                   PRP4      13
      DIMENSION LBID(2)                                                 PRP4      14
      DATA LBID/5,20/                                                   PRP4      15
      CALL BPNFO(ITRN,ID)                                               PRP4      16
      J=2                                                               PRP4      17
      ICRIT(J) = ID(3)                                                  PRP4      18
C                                                                       PRP4      19
C  A VALUE OF ICRIT(J)=2 IS ALLOWED FOR THIS VARIABLE, OTHERWISE THE    PRP4      20
C   PROGRAM AWAITS ANOTHER ATTENTION SOURCE                             PRP4      21
      IF(ICRIT(J).EQ.1) CALL WAITE(DUM,0,DUM,DUM)                       PRP4      22
      CALL ASCAL(1)                                                     PRP4      23
      CALL ERASE(NALL)                                                  PRP4      24
      CALL ENLB(2,LBID)                                                 PRP4      25
      CALL GENDF(NAIRFL,0)                                              PRP4      26
      CALL GENDF(NCAFL2,0)                                              PRP4      27
      CALL GENDF(NUPS,0)                                                PRP4      28
      CALL GENDF(LRAFL2,0)                                              PRP4      29
      CALL GENDF(NMXB,0)                                                PRP4      30
      CALL WAITE(DUM,0,DUM,DUM)                                         PRP4      31
      END                                                               PRP4      32
```

```
      OVERLAY(15,0)                                                     AFL2       2
      PROGRAM AFL2                                                      AFL2       3
C                                                                       AFL2       4
C     THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE IOLWRIN              AFL2       5
C                                                                       AFL2       6
      COMMON/OUTCOM/                                                    OUTCOM     2
     1    X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2               OUTCOM     3
      COMMON/ICNTRL/J          ,ICRIT(2)  ,LL(2)     ,IGO(2)            ICNTRL     2
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1    ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2    ,LRID(6)   ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6) NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)  NPRCD      3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)    NPRCD      4
     3    ,NAF1(6)   ,NAF2(6)                                           NPRCD      5
      COMMON/NCHARS/NNEQ       ,NAEQ      ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2) NCHARS     2
     1    ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)   NCHARS     3
     2    ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)    ,NMACHQ(2),NALPHA(2) NCHARS     4
     3    ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI      ,FMTF                NCHARS     5
      COMMON C       ,CK       ,RS       ,FM       ,ALPHA               COMMON     2
      COMMON/AINPUT/AIN(24)    ,NNI(7)    ,HI(6)                        AINPUT     2
      COMMON/YUVSAV/NNN(3)     ,YUV(156)                                YUVSAV     2
      COMMON/PTARFL/XX(40,2)   ,YY(40,2)  ,AM(40,2)  ,CA        ,SA     PTARFL     2
      COMMON/RBUBCM/RBUB       ,UBINIT    ,IRBUB                        RBUBCM     2
      COMMON/COMPRS/XP(160,2),PP(160,2),NP(2)                           COMPRS     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      CALL ASCAL(1)                                                     AFL2      19
      CALL ERASG(IDSCAL)                                                AFL2      20
      CALL ERASG(IDSCAL-1)                                              AFL2      21
      CALL ERASE(LRNOGO)                                                AFL2      22
C                                                                       AFL2      23
C     ERASE TEXT ENTITIES PREVIOUSLY DISPLAYED BY THIS PROGRAM          AFL2      24
      CALL ERASE(LRRUEQ)                                                AFL2      25
      CALL ERASE(LRID)                                                  AFL2      26
      CALL ERASE(LRUBEQ)                                                AFL2      27
      CALL ERASE(LRPOEQ)                                                AFL2      28
      IICRIT = ICRIT(J)                                                 AFL2      29
      L = LL(J)                                                         AFL2      30
      CALL IOLWRIN(IICRIT,L)                                            AFL2      31
      IF(NN2.NE.1) CALL GENDF(LRNOGO,0)                                 AFL2      32
C                                                                       AFL2      33
C     DISPLAY THE OUTPUT VALUES OF RBUB AND UB                          AFL2      34
      CALL ENSHFT(NRUEQ,5,RBUB,FMTF)                                    AFL2      35
      CALL MODFY(LRRUEQ,1,2,NRUEQ)                                      AFL2      36
      CALL GENDF(LRRUEQ,0)                                              AFL2      37
      IF(IRBUB.EQ.0) GO TO 4                                            AFL2      38
      IF(IRBUB.EQ.2) CALL GENDF(LRID,0)                                 AFL2      39
      GO TO 6                                                           AFL2      40
    4 CALL ENSHFT(NUBEQ,3,UBINIT,FMTF)                                  AFL2      41
      CALL MODFY(LRUBEQ,1,2,NUBEQ)                                      AFL2      42
      CALL GENDF(LRUBEQ,0)                                              AFL2      43
    6 CONTINUE                                                          AFL2      44
      IF(NN2.EQ.0) GO TO 8                                              AFL2      45
      NN = NP(J)                                                        AFL2      46
C                                                                       AFL2      47
C     DEPENDING ON THE VALUE OF NN2, DISPLAY THE OUTPUT VALUE OF PO     AFL2      48
      CALL ENSHFT(NPOEQ,3,PP(NN,J),FMTF)                                AFL2      49
      CALL MODFY(LRPOEQ,1,2,NPOEQ)                                      AFL2      50
      CALL GENDF(LRPOEQ,0)                                              AFL2      51
C                                                                       AFL2      52
C     DEPENDING ON THE VALUE OF NN2 AND IGO(J), DISPLAY ASTERISKS       AFL2      53
C     INDICATING THE NEXT PROGRAM STEP                                  AFL2      54
      IF(NN2.EQ.1) IGO(J)=1                                             AFL2      55
      IF(NN2.EQ.1) CALL GENDF(NPAFU1,0)                                 AFL2      56
      IF(IGO(1).EQ.1.AND.IGO(2).EQ.1) CALL GENDF(NPDWN1,0)              AFL2      57
    8 CALL PLOTT(0.0,1.0,0.0,1.0)                                       AFL2      58
C                                                                       AFL2      59
C     WAIT FOR AN ATTENTION SOURCE                                      AFL2      60
      CALL WAITE(DUM,0,DUM,DUM)                                         AFL2      61
      END                                                               AFL2      62
```

```
      OVERLAY(16,0)                                                     PRP5       2
      PROGRAM PRP5                                                      PRP5       3
C                                                                       PRP5       4
C   THIS PROGRAM DISPLAYS ITEMS NEEDED FOR PROGRAM AFU3                 PRP5       5
C                                                                       PRP5       6
      COMMON/COMNXY/NXY1(6)     ,NXY2(6)                                COMNXY     2
      COMMON/ICNTRL/J           ,ICRIT(2) ,LL(2)      ,IGO(2)           ICNTRL     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LROEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDDQ3(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      DIMENSION LBID(2)                                                 PRP5      13
      DATA LBID/6,20/                                                   PRP5      14
      J = 1                                                             PRP5      15
      IGO(J) = 0                                                        PRP5      16
      CALL ASCAL(1)                                                     PRP5      17
      CALL ERASE(NALL)                                                  PRP5      18
      CALL ENLB(2,LBID)                                                 PRP5      19
      CALL GENDF(NAIRFL,0)                                              PRP5      20
      CALL GENDF(NCAFU3,0)                                              PRP5      21
      CALL GENDF(NPAFU2,0)                                              PRP5      22
      CALL GENDF(NUPS,0)                                                PRP5      23
      CALL GENDF(LRAFU3,0)                                              PRP5      24
      CALL GENDF(NMXB,0)                                                PRP5      25
      CALL WAITE(DUM,0,DUM,DUM)                                         PRP5      26
      END                                                               PRP5      27
```

```
      OVERLAY(17,0)                                                     AFU3       2
      PROGRAM AFU3                                                      AFU3       3
C                                                                       AFU3       4
C   THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE IOSPCT2                AFU3       5
C                                                                       AFU3       6
      COMMON/OUTCOM/                                                    OUTCOM     2
     1    X1(160)    ,Y1(160)    ,Y2(160)    ,NN1        ,NN2           OUTCOM     3
      COMMON/ICNTRL/J            ,ICRIT(2)   ,LL(2)      ,IGO(2)        ICNTRL     2
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1    ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2    ,LRID(6)   ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)  NOUT       4
      COMMON/NPRCO/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCO      2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)  NPRCO      3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)    NPRCO      4
     3    ,NAF1(6)   ,NAF2(6)                                           NPRCO      5
      COMMON/NCHARS/NNEQ         ,NAEQ       ,NXDOEQ(2),NDVIEQ(2),NXSEQ(2)  NCHARS 2
     1    ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)   NCHARS     3
     2    ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)    ,NMACHQ(2),NALPHA(2) NCHARS     4
     3    ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI      ,FMTF                NCHARS     5
      COMMON C      ,CK        ,RS        ,FM        ,ALPHA             COMMON     2
      COMMON/AINPUT/AIN(24)    ,NNI(7)     ,HI(6)                       AINPUT     2
      COMMON/YUVSAV/NNN(3)     ,YUV(156)                                YUVSAV     2
      COMMON/PTARFL/XX(40,2) ,YY(40,2) ,AM(40,2) ,CA        ,SA         PTARFL     2
      COMMON/COMPRS/XP(160,2),PP(160,2),NP(2)                           COMPRS     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      COMMON/COMSPR/ARMO(160)                                           AFU3      18
      CALL ASCAL(1)                                                     AFU3      19
      CALL ERASE(IDSCAL)                                                AFU3      20
      CALL ERASE(IDSCAL-1)                                              AFU3      21
      CALL ERASE(LRNOGO)                                                AFU3      22
      CALL ERASE(LRPOEQ)                                                AFU3      23
      L = LL(J)                                                         AFU3      24
      CALL IOSPCT2(J,L)                                                 AFU3      25
C                                                                       AFU3      26
C   DEPENDING ON THE VALUE OF NN2 AND IGO(J), DISPLAY ASTERISKS         AFU3      27
C   INDICATING THE NEXT PROGRAM STEP                                    AFU3      28
      IF(NN2.NE.1) CALL GENDF(LRNOGO,0)                                 AFU3      29
      IF(NN2.EQ.1) IGO(J) = 1                                           AFU3      30
      IF(NN2.EQ.1) CALL GENDF(NPAFL1,0)                                 AFU3      31
      IF(IGO(1).EQ.1.AND.IGO(2).EQ.1) CALL GENDF(NPDWN1,0)              AFU3      32
      IF(NN2.EQ.0) GO TO 8                                              AFU3      33
      NN = NP(J)                                                        AFU3      34
      CALL ENSHFT(NPOEQ,3,PP(NN,J),FMTF)                                AFU3      35
      CALL MODFY(LRPOEQ,1,2,NPOEQ)                                      AFU3      36
      CALL GENDF(LRPOEQ,0)                                              AFU3      37
    8 DO 10 I=1,160                                                     AFU3      38
   10 Y1(I) = ARMO(I)                                                   AFU3      39
      CALL PLOTT1(0.0,1.0,0.0,1.0)                                      AFU3      40
C                                                                       AFU3      41
C   WAIT FOR AN ATTENTION SOURCE                                        AFU3      42
      CALL WAITF(DUM,0,DUM,DUM)                                         AFU3      43
      END                                                               AFU3      44
```

```
      OVERLAY(20,0)                                                      PRP6      2
      PROGRAM PRP6                                                       PRP6      3
C                                                                        PRP6      4
C     THIS PROGRAM DISPLAYS ITEMS NEEDED FOR PROGRAM DWN1                PRP6      5
C                                                                        PRP6      6
      COMMON/ICNTRL/J         ,ICRIT(2) ,LL(2)     ,IGO(2)               ICNTRL    2
      COMMON/OUTCOM/                                                     OUTCOM    2
     1    X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2                OUTCOM    3
      COMMON/INPUT/                                                      INPUT     2
     1     LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT     3
     2    ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT     4
     3    ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT     5
     4    ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT     6
     5    ,NLGRNG(6),NPARAB(6)                                           INPUT     7
      COMMON/NOUT/ NAIRFL(6)                                             NOUT      2
     1    ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT      3
     2    ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT      4
      COMMON/NAXES/ NALL(6)                                              NAXES     2
     1    ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES     3
     2    ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES     4
     3    ,NDU1B(6) ,NDU2B(6) ,NDDQ3(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES     5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD     2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD     3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD     4
     3    ,NAF1(6)  ,NAF2(6)                                             NPRCD     5
      DIMENSION LBID(2)                                                  PRP6     13
      DATA LBID/7,20/                                                    PRP6     14
      CALL ASCAL(1)                                                      PRP6     15
      CALL ERASE(NALL)                                                   PRP6     16
      CALL ENLB(2,LBID)                                                  PRP6     17
      CALL GENDF(NAIRFL,0)                                               PRP6     18
      CALL GENDF(NCDWN1,0)                                               PRP6     19
      CALL GENDF(NPAFU3,0)                                               PRP6     20
      CALL GENDF(NPAFL2,0)                                               PRP6     21
      CALL GENDF(LRNN6,0)                                                PRP6     22
      CALL GENDF(NDWNB,0)                                                PRP6     23
      CALL WAITE(DUM,0,DUM,DUM)                                          PRP6     24
      END                                                                PRP6     25
```

```
      OVERLAY(21,0)                                                     DWN1       2
      PROGRAM DWN1                                                      DWN1       3
C                                                                       DWN1       4
C   THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE IODNSTM                DWN1       5
C                                                                       DWN1       6
      COMMON/ICNTRL/J          ,ICRIT(2) ,LL(2)      ,IGO(2)            ICNTRL     2
      COMMON/OUTCOM/                                                    OUTCOM     2
     1    X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2               OUTCOM     3
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1    ,LRDEEQ(6),LRVSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)  NOUT       3
     2    ,LRID(6)   ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)  NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1    ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)  NPRCD      3
     2    ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)    NPRCD      4
     3    ,NAF1(6)   ,NAF2(6)                                           NPRCD      5
      COMMON C       ,CK        ,RS        ,FM        ,ALPHA            COMMON     2
      COMMON/AINPUT/AIN(24)    ,NNI(7)    ,HI(6)                        AINPUT     2
      COMMON/YUVSAV/NNN(3)     ,YUV(156)                                YUVSAV     2
      COMMON/PTARFL/XX(40,2)  ,YY(40,2)  ,AM(40,2)  ,CA        ,SA      PTARFL     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      CALL ASCAL(1)                                                     DWN1      16
      CALL ERASG(IDSCAL)                                                DWN1      17
      CALL ERASG(IDSCAL-1)                                              DWN1      18
      CALL ERASE(LRNOGO)                                                DWN1      19
      CALL IODNSTM(J)                                                   DWN1      20
      IF(NN2.NE.1) CALL GENDF(LRNOGO,0)                                 DWN1      21
      IF(NN2.EQ.1) CALL GENDF(NPDWN2,0)                                 DWN1      22
      CALL PLOTT2(1.0,10.0,0.8,1.0,0.8,1.0,2)                           DWN1      23
C                                                                       DWN1      24
C   WAIT FOR AN ATTENTION SOURCE                                        DWN1      25
      CALL WAITE(DUM,0,DUM,DUM)                                         DWN1      26
      END                                                               DWN1      27
```

```
      OVERLAY(22,0)                                                     DWN2       2
      PROGRAM DWN2                                                      DWN2       3
C                                                                       DWN2       4
C   THIS PROGRAM DISPLAYS OUTPUT FROM SUBROUTINE AKUTTA                 DWN2       5
C                                                                       DWN2       6
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1   ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)   NOUT       3
     2   ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)   NOUT       4
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1   ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)    NAXES      3
     2   ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NMO1B(6) ,NMO2B(6)    NAXES      4
     3   ,NDU1B(6) ,NDU2B(6) ,NDOQB(6) ,NPOB(6)  ,NP1B(6)  ,NPKTAB(6)   NAXES      5
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1   ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)   NPRCD      3
     2   ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)     NPRCD      4
     3   ,NAF1(6)  ,NAF2(6)                                             NPRCD      5
      COMMON/COMPRS/XP(160,2),PP(160,2),NP(2)                           COMPRS     2
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      CALL ASCAL(1)                                                     DWN2      14
      CALL ERASG(IDSCAL)                                                DWN2      15
      CALL ERASG(IDSCAL-1)                                              DWN2      16
      CALL ERASE(NALL)                                                  DWN2      17
      CALL ENLB(1,20)                                                   DWN2      18
      CALL GENDF(NAIRFL,0)                                              DWN2      19
      CALL GENDF(NCDWN2,0)                                              DWN2      20
      CALL GENDF(NAF1,0)                                                DWN2      21
      CALL GENDF(NKTAB,0)                                               DWN2      22
      CALL AKUTTA                                                       DWN2      23
      PMAX = PP(1,1)                                                    DWN2      24
      PMIN = PP(1,1)                                                    DWN2      25
      XMAX = 1.0                                                        DWN2      26
      XMIN = XP(1,1)                                                    DWN2      27
      IF(XP(1,2).LT.XMIN) XMIN = XP(1,2)                                DWN2      28
      DO 20 J=1,2                                                       DWN2      29
      NN = NP(J)                                                        DWN2      30
      DO 10 I=1,NN                                                      DWN2      31
      IF(PP(I,J) -PMAX) 6,5,4                                           DWN2      32
    4 PMAX = PP(I,J)                                                    DWN2      33
      GO TO 10                                                          DWN2      34
    5 IF(PP(I,J)-PMIN) 8,10,10                                          DWN2      35
    8 PMIN = PP(I,J)                                                    DWN2      36
   10 CONTINUE                                                          DWN2      37
   20 CONTINUE                                                          DWN2      38
      IF(PMIN.GT.0.0) PMIN =0.0                                         DWN2      39
      CALL AREA1(XMIN,XMAX,PMIN,PMAX)                                   DWN2      40
      NGRAF = 0                                                         DWN2      41
      NXY1(5) = 0                                                       DWN2      42
      CALL DLETE(NXY1)                                                  DWN2      43
      DO 40 J=1,2                                                       DWN2      44
      NN = NP(J)                                                        DWN2      45
      IF(NN.GE.60) NGRAF=1                                              DWN2      46
      IF(NN.GE.120) NGRAF=2                                             DWN2      47
      IF(NGRAF.EQ.0) GO TO 36                                           DWN2      48
      NXY1(6) = J                                                       DWN2      49
      DO 30 I=1,NGRAF                                                   DWN2      50
      NXY1(5) = I                                                       DWN2      51
   30 CALL PLYLN(NXY1,1,XP(60*NGRAF-59,J),PP(60*NGRAF-59,J),60)         DWN2      52
   36 NXY1(6) = J                                                       DWN2      53
      NXY1(5) = 60                                                      DWN2      54
      CALL PLYLN(NXY1,1,XP(1+60*NGRAF,J),PP(1+60*NGRAF,J),              DWN2      55
     1    NP(J)-60*NGRAF-1)                                             DWN2      56
   40 CONTINUE                                                          DWN2      57
```

```
      NXY1(5) = 0
      NXY1(6) = 0
      CALL GENDF(NXY1,0)
C
C   WAIT FOR AN ATTENTION SOURCE
      CALL WAITE(DUM,0,DUM,DUM)
      END
```

```
      OVERLAY(23,0)                                                     CVLI       2
      PROGRAM CVLI                                                      CVLI       3
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARA3(6)                                           INPUT      7
      COMMON/NCHARS/NNEQ     ,NAEQ     ,NXOOEQ(2),NDVIEQ(2),NXSEQ(2)    NCHARS     2
     1   ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)    NCHARS     3
     2   ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)   ,NMACHQ(2),NALPHA(2)   NCHARS     4
     3   ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI     ,FMTF                  NCHARS     5
      COMMON/AINPUT/AIN(24)  ,NNI(7)   ,HI(6)                           AINPUT     2
      DIMENSION DUM(6),ID(6)                                            CVLI       7
C                                                                       CVLI       8
C   RETRIEVE ATTENTION INFORMATION FROM THE TEXT ENTITY IN A SIX INTEGER CVLI      9
C   ARRAY ID                                                            CVLI      10
      CALL BPNFO(ITRN,ID)                                               CVLI      11
C                                                                       CVLI      12
C   ERASE THE TEXT ENTITY                                               CVLI      13
      CALL ERASE(ID)                                                    CVLI      14
C                                                                       CVLI      15
C   REPLACE THE TEXT ENTITY WITH A CORRESPONDING LIGHT REGISTER         CVLI      16
      CALL ENLR(1,ID)                                                   CVLI      17
C                                                                       CVLI      18
C   DISPLAY THE NUMBERS BEING TYPED INTO THE LIGHT REGISTER FROM THE    CVLI      19
C   KEYBOARD                                                            CVLI      20
      CALL KBNPT(ID,IVAL)                                               CVLI      21
      CALL ASCAL(1)                                                     CVLI      22
C                                                                       CVLI      23
C   ERASE THE LIGHT REGISTER                                            CVLI      24
      CALL ENLR(0,ID)                                                   CVLI      25
      ID1 = ID(1)-20                                                    CVLI      26
C                                                                       CVLI      27
C   PUT THE NEW VALUE INTO THE CORRESPONDING TEXT ENTITY                CVLI      28
      GO TO (10,20,30,40,50,60),ID1                                     CVLI      29
   10 NNI(1) = IVAL                                                     CVLI      30
      CALL ENSHFT(NNEQ,3,NNI(1),FMTI)                                   CVLI      31
      CALL MODFY(ID,1,1,NNEQ)                                           CVLI      32
      GO TO 200                                                         CVLI      33
   20 NNI(2) = IVAL                                                     CVLI      34
      CALL ENSHFT(NAEQ,3,NNI(2),FMTI)                                   CVLI      35
      CALL MODFY(ID,1,1,NAEQ)                                           CVLI      36
      GO TO 200                                                         CVLI      37
   30 NNI(3) = IVAL                                                     CVLI      38
      CALL ENSHFT(NNEQ,3,NNI(3),FMTI)                                   CVLI      39
      CALL MODFY(ID,1,1,NNEQ)                                           CVLI      40
      GO TO 200                                                         CVLI      41
   40 NNI(4) = IVAL                                                     CVLI      42
      CALL ENSHFT(NNEQ,3,NNI(4),FMTI)                                   CVLI      43
      CALL MODFY(ID,1,1,NNEQ)                                           CVLI      44
      GO TO 200                                                         CVLI      45
   50 NNI(5) = IVAL                                                     CVLI      46
      CALL ENSHFT(NNEQ,3,NNI(5),FMTI)                                   CVLI      47
      CALL MODFY(ID,1,1,NNEQ)                                           CVLI      48
      GO TO 200                                                         CVLI      49
   60 NNI(6) = IVAL                                                     CVLI      50
      CALL ENSHFT(NNEQ,3,NNI(6),FMTI)                                   CVLI      51
      CALL MODFY(ID,1,1,NNEQ)                                           CVLI      52
C                                                                       CVLI      53
C   DISPLAY THIS TEXT ENTITY WHICH HAS BEEN CHANGED                     CVLI      54
  200 CALL GENDF(ID,0)                                                  CVLI      55
C                                                                       CVLI      56
C   WAIT FOR AN ATTENTION SOURCE                                        CVLI      57
      CALL WAITE(DUM,0,DUM,DUM)                                         CVLI      58
      END                                                               CVLI      59
```

```
      OVERLAY(24,0)                                                     CVLR       2
      PROGRAM CVLR                                                      CVLR       3
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRX00Q(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      COMMON/NCHARS/NNEQ      ,NAEQ      ,NX00EQ(2),NDVIEQ(2),NXSEQ(2)  NCHARS     2
     1   ,NYSOEQ(2),NXAEQ(2) ,NCYDEQ(2),NSLEQ(2) ,NDEEQ(2) ,NYSEQ(2)    NCHARS     3
     2   ,NDVOEQ(2),NRUEQ(2) ,NUBEQ(2) ,NID(2)    ,NMACHQ(2),NALPHA(2)  NCHARS     4
     3   ,NYIUEQ(2),NYILEQ(2),NPOEQ(2) ,FMTI      ,FMTF                 NCHARS     5
      COMMON/AINPUT/AIN(24)   ,NNI(7)    ,HI(6)                         AINPUT     2
      COMMON C       ,CK       ,RS        ,FM        ,ALPHA              COMMON     2
      DIMENSION DUM(6),ID(6)                                            CVLR       8
C                                                                       CVLR       9
C     RETRIEVE ATTENTION INFORMATION FROM THE TEXT ENTITY IN A SIX INTEGER CVLR   10
C     ARRAY ID                                                          CVLR      11
      CALL BPNFO(ITRN,ID)                                               CVLR      12
C                                                                       CVLR      13
C     ERASE THE TEXT ENTITY                                             CVLR      14
      CALL ERASE(ID)                                                    CVLR      15
C                                                                       CVLR      16
C     REPLACE THE TEXT ENTITY WITH A CORRESPONDING LIGHT REGISTER       CVLR      17
      CALL ENLR(1,ID)                                                   CVLR      18
C                                                                       CVLR      19
C     DISPLAY THE NUMBERS BEING TYPED INTO THE LIGHT REGISTER FROM THE  CVLR      20
C     KEYBOARD                                                          CVLR      21
      CALL KBNPT(ID,VAL)                                                CVLR      22
C                                                                       CVLR      23
C     ERASE THE LIGHT REGISTER                                          CVLR      24
      CALL ENLR(0,ID)                                                   CVLR      25
      CALL ASCAL(1)                                                     CVLR      26
      ID1 = ID(1)                                                       CVLR      27
C                                                                       CVLR      28
C     PUT THE NEW VALUE INTO THE CORRESPONDING TEXT ENTITY              CVLR      29
      GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130,140),ID1        CVLR      30
   10 AIN(8) = VAL                                                      CVLR      31
      CALL ENSHFT(NX00EQ,4,AIN(8),FMTF)                                 CVLR      32
      CALL MODFY(ID,1,2,NX00EQ)                                         CVLR      33
      GO TO 200                                                         CVLR      34
   20 AIN(1) = VAL                                                      CVLR      35
      CALL ENSHFT(NDVIEQ,8,AIN(1),FMTF)                                 CVLR      36
      CALL MODFY(ID,1,2,NDVIEQ)                                         CVLR      37
      GO TO 200                                                         CVLR      38
   30 AIN(2) = VAL                                                      CVLR      39
      CALL ENSHFT(NXSEQ,3,AIN(2),FMTF)                                  CVLR      40
      CALL MODFY(ID,1,2,NXSEQ)                                          CVLR      41
      GO TO 200                                                         CVLR      42
   40 AIN(3) = VAL                                                      CVLR      43
      CALL ENSHFT(NXAEQ,3,AIN(3),FMTF)                                  CVLR      44
      CALL MODFY(ID,1,2,NXAEQ)                                          CVLR      45
      GO TO 200                                                         CVLR      46
   50 AIN(4) = VAL                                                      CVLR      47
      CALL ENSHFT(NCYDEQ,4,AIN(4),FMTF)                                 CVLR      48
      CALL MODFY(ID,1,2,NCYDEQ)                                         CVLR      49
      GO TO 200                                                         CVLR      50
   60 AIN(5) = VAL                                                      CVLR      51
      CALL ENSHFT(NXAEQ,3,AIN(5),FMTF)                                  CVLR      52
      CALL MODFY(ID,1,2,NXAEQ)                                          CVLR      53
      GO TO 200                                                         CVLR      54
   70 AIN(6) = VAL                                                      CVLR      55
      CALL ENSHFT(NCYDEQ,4,AIN(6),FMTF)                                 CVLR      56
      CALL MODFY(ID,1,2,NCYDEQ)                                         CVLR      57
      GO TO 200                                                         CVLR      58
   80 AIN(7) = VAL                                                      CVLR      59
      CALL ENSHFT(NSLEQ,10,AIN(7),FMTF)                                 CVLR      60
      CALL MODFY(ID,1,2,NSLEQ)                                          CVLR      61
   90 GO TO 200                                                         CVLR      62
```

```
      100 GO TO 200                                                     CVLR    63
      110 FM = VAL                                                      CVLR    64
          CALL ENSHFT(NMACHQ,9,FM,FMTF)                                 CVLR    65
          CALL MODFY(ID,1,2,NMACHQ)                                     CVLR    66
          GO TO 200                                                     CVLR    67
      120 ALPHA = VAL                                                   CVLR    68
          CALL ENSHFT(NALPHA,6,ALPHA,FMTF)                              CVLR    69
          CALL MODFY(ID,1,2,NALPHA)                                     CVLR    70
          GO TO 200                                                     CVLR    71
      130 AIN(11) = VAL                                                 CVLR    72
          CALL ENSHFT(NYIUEQ,10,AIN(11),FMTF)                           CVLR    73
          CALL MODFY(ID,1,2,NYIUEQ)                                     CVLR    74
          GO TO 200                                                     CVLR    75
      140 AIN(12) = VAL                                                 CVLR    76
          CALL ENSHFT(NYILEQ,10,AIN(12),FMTF)                           CVLR    77
          CALL MODFY(ID,1,2,NYILEQ)                                     CVLR    78
C                                                                       CVLR    79
C     DISPLAY THIS TEXT ENTITY WHICH HAS BEEN CHANGED                   CVLR    80
      200 CALL GENOF(ID,0)                                              CVLR    81
C                                                                       CVLR    82
C     WAIT FOR AN ATTENTION SOURCE                                      CVLR    83
          CALL WAITE(DUM,0,DUM,DUM)                                     CVLR    84
          END                                                           CVLR    85
```

```
      OVERLAY(25,0)                                                     CHGV       2
      PROGRAM CHGV                                                      CHGV       3
      COMMON/ICNTRL/J       ,ICRIT(2) ,LL(2)    ,IGO(2)                 ICNTRL     2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2   ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)   INPUT      4
     3   ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)   INPUT      5
     4   ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)    INPUT      6
     5   ,NLGRNG(6),NPARAB(6)                                           INPUT      7
      LL(J) = LL(J)+1                                                   CHGV       6
      IF(LL(J).EQ.2) CALL ERASE(NLGRNG,0)                               CHGV       7
      IF(LL(J).EQ.3) CALL ERASE(NPARAB,0)                               CHGV       8
      IF(LL(J).EQ.2) CALL GENDF(NPARAB,0)                               CHGV       9
      CALL WAITE(DUM,0,DUM,DUM)                                         CHGV      10
      END                                                               CHGV      11
```

```
      OVERLAY(25,0)                                                     STOP       2
      PROGRAM STOP                                                      STOP       3
      COMMON/NCON/  ICON                                                NCON       2
      COMMON/INPUT/                                                     INPUT      2
     1    LRUPS(6) ,LRSTG(6) ,LRAFU2(6),LRAFL2(6),LRAFU3(6),LRXOOQ(6)   INPUT      3
     2  ,LRDIEQ(6),LRXSEQ(6),LRXAUP(6),LRCYDU(6),LRXALW(6),LRCYDL(6)    INPUT      4
     3  ,LRSLEQ(6),LRMACH(6),LRALFA(6),LRYIU(6) ,LRYIL(6) ,LRSTRT(6)    INPUT      5
     4  ,LRNN1(6) ,LRNA2(6) ,LRNN3(6) ,LRNN4(6) ,LRNN5(6) ,LRNN6(6)     INPUT      6
     5  ,NLGRNG(6),NPARAB(6)                                            INPUT      7
      COMMON/NOUT/ NAIRFL(6)                                            NOUT       2
     1  ,LRDEEQ(6),LRYSOQ(6),LRYSEQ(6),LRDOEQ(6),LRRUEQ(6),LRUBEQ(6)    NOUT       3
     2  ,LRID(6)  ,LRPOEQ(6),LRNOGO(6),LRSUB(6) ,LRSUPR(6),LRFLOW(6)    NOUT       4
      COMMON/NPRCD/ NCUPS1(6),NCUPS2(6),NCAFU2(6),NCAFL2(6),NCAFU3(6)   NPRCD      2
     1  ,NCDWN1(6),NCDWN2(6),NPUPS1(6),NPUPS2(6),NPAFU1(6),NPAFU2(6)    NPRCD      3
     2  ,NPAFU3(6),NPAFL1(6),NPAFL2(6),NPDWN1(6),NPDWN2(6),NUPS(6)      NPRCD      4
     3  ,NAF1(6)  ,NAF2(6)                                              NPRCD      5
      COMMON/NAXES/ NALL(6)                                             NAXES      2
     1  ,NMXB(6)  ,NUPB(6)  ,NDUDXB(6),NAF3B(6) ,NDWNB(6) ,NKTAB(6)     NAXES      3
     2  ,NX1B(6)  ,NX2B(6)  ,NYB(6)   ,NMB(6)   ,NM01B(6) ,NM02B(6)     NAXES      4
     3  ,NDU1B(6) ,NDU2B(6) ,NDDQ3(6) ,NPOB(6)  ,NP13(6)  ,NPKTAB(6)    NAXES      5
C                                                                       STOP       9
C   ERASE THE SCREEN DISPLAYS                                           STOP      10
      CALL ASCAL(1)                                                     STOP      11
      CALL ENLB(0,1)                                                    STOP      12
      CALL ERASG(1)                                                     STOP      13
      CALL ERASG(2)                                                     STOP      14
      CALL ERASE(NALL)                                                  STOP      15
C                                                                       STOP      16
C   CLOSE THE DATA FILE                                                 STOP      17
      CALL CLOSF                                                        STOP      18
C                                                                       STOP      19
C   RELEASE THE CONSOLE                                                 STOP      20
      CALL RLCON(ICON)                                                  STOP      21
      END                                                               STOP      22
```

```
      SUBROUTINE PLOTT(X1MIN,X1MAX,Y1MIN,Y1MAX)                         PLOTT      2
C                                                                       PLOTT      3
C     THIS SUBROUTINE DISPLAYS TWO DISJOINTED CURVES COVERING THE       PLOTT      4
C     GRAPHICAL DISPLAY AREA                                            PLOTT      5
C                                                                       PLOTT      6
      COMMON/OUTCOM/                                                    OUTCOM     2
     1   X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2                OUTCOM     3
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY     2
      COMMON/AINPUT/AIN(24)   ,NNI(7)    ,HI(6)                         AINPUT     2
C                                                                       PLOTT     10
C     IF THERE IS ONLY ONE POINT TO BE PLOTTED, FORGET IT AND PLOT FOUR PLOTT     11
C     POINTS ON THE SCREEN TO FORM A LARGE X COVERING THE SCREEN        PLOTT     12
      IF(NN1.GT.1) GO TO 5                                              PLOTT     13
      NN1 = 4                                                           PLOTT     14
      X1(1) = X1MIN                                                     PLOTT     15
      X1(2) = X1MAX                                                     PLOTT     16
      X1(3) = X1MIN                                                     PLOTT     17
      X1(4) = X1MAX                                                     PLOTT     18
      Y1(1) = Y1MIN                                                     PLOTT     19
      Y1(2) = Y1MAX                                                     PLOTT     20
      Y1(3) = Y1MAX                                                     PLOTT     21
      Y1(4) = Y1MIN                                                     PLOTT     22
      XMIN = X1MIN                                                      PLOTT     23
      YMIN = Y1MIN                                                      PLOTT     24
      XMAX = X1MAX                                                      PLOTT     25
      YMAX = Y1MAX                                                      PLOTT     26
      NXY1(5) = 0                                                       PLOTT     27
      CALL DLETE(NXY1)                                                  PLOTT     28
      NXY1(5) = 60                                                      PLOTT     29
      CALL AREA1(XMIN,XMAX,YMIN,YMAX)                                   PLOTT     30
      CALL PLYLN(NXY1,1,X1(1),Y1(1),3)                                  PLOTT     31
      CALL GENDF(NXY1,0)                                                PLOTT     32
      RETURN                                                            PLOTT     33
C                                                                       PLOTT     34
C     FIND THE LARGEST AND SMALLEST VALUES OF Y1                        PLOTT     35
    5 CALL AMXMN1(YMAX,YMIN)                                            PLOTT     36
      NXY1(5) = 0                                                       PLOTT     37
      CALL DLETE(NXY1)                                                  PLOTT     38
      XMIN = X1(1)                                                      PLOTT     39
      XMAX = X1(NN1)                                                    PLOTT     40
      IF(X1MIN.LT.XMIN) XMIN=X1MIN                                      PLOTT     41
      IF(X1MAX.GT.XMAX) XMAX = X1MAX                                    PLOTT     42
      IF(Y1MIN.LT.YMIN) YMIN = Y1MIN                                    PLOTT     43
      IF(Y1MAX.GT.YMAX) YMAX = Y1MAX                                    PLOTT     44
C                                                                       PLOTT     45
C     FIND THE NUMBER OF POINTS IN THE FIRST AND SECOND CURVES          PLOTT     46
      NNN1 = NNI(7)-1                                                   PLOTT     47
      NNN2 = NN1-NNI(7)                                                 PLOTT     48
      IF(NNI(7).EQ.0) NNN1=NN1                                          PLOTT     49
      IF(NNI(7).EQ.0) NNN2=0                                            PLOTT     50
C                                                                       PLOTT     51
C     CREATE THE POLYLINE ENTITY FOR THE FIRST CURVE                    PLOTT     52
      NGRAF1 = 0                                                        PLOTT     53
      IF(NNN1.GT.60) NGRAF1 = 1                                         PLOTT     54
      IF(NNN1.GT.120) NGRAF1 = 2                                        PLOTT     55
      IF(NGRAF1.EQ.0) GO TO 30                                          PLOTT     56
      DO 20 I=1,NGRAF1                                                  PLOTT     57
      NXY1(5) = I                                                       PLOTT     58
   20 CALL PLYLN(NXY1,1,X1(60*I-59),Y1(60*I-59),60)                     PLOTT     59
   30 IF((NNN1-60*NGRAF1-1).LE.0) GO TO 40                              PLOTT     60
      NXY1(5) = NGRAF1+1                                                PLOTT     61
      CALL PLYLN(NXY1,1,X1(1+60*NGRAF1),Y1(1+60*NGRAF1),NNN1-60*NGRAF1-1PLOTT     62
     1)                                                                 PLOTT     63
C                                                                       PLOTT     64
C     CREATE THE POLYLINE ENTITY FOR THE SECOND CURVE                   PLOTT     65
   40 NGRAF2=0                                                          PLOTT     66
      IF(NNN2.GT.60) NGRAF2 = 1                                         PLOTT     67
      IF(NNN2.GT.120) NGRAF2 = 2                                        PLOTT     68
      N1 = NGRAF1+2                                                     PLOTT     69
```

```
      IF(NGRAF2.EQ.0) GO TO 60
      N2 = N1+NGRAF2-1
      DO 50 I=N1,N2
      NXY1(5) = I
      NNNN1 = NNN1+60*(I-N1+1)-58
   50 CALL PLYLN(NXY1,1,X1(NNNN1),Y1(NNNN1),60)
   60 IF((NNN2-60*NGRAF2-1).LE.0) GO TO 70
      NXY1(5) = N1+NGRAF2
      CALL PLYLN(NXY1,1,X1(NNN1+60*NGRAF2+2),Y1(NNN1+60*NGRAF2+2),
     1          NNN2-60*NGRAF2-1)
C
C   CREATE THE GRID DISPLAY
   70 CALL AREA1(XMIN,XMAX,YMIN,YMAX)
      NXY1(5) = 0
C
C   DISPLAY THE TWO POLYLINE ENTITIES
      CALL GENDF(NXY1,0)
      RETURN
      END
```

```
      SUBROUTINE PLOTT1(X1MIN,X1MAX,Y1MIN,Y1MAX)                        PLOTT1   2
C                                                                       PLOTT1   3
C    THIS SUBROUTINE DISPLAYS ONE CURVE IN THE GRAPHIC DISPLAY AREA     PLOTT1   4
      COMMON/OUTCOM/                                                    OUTCOM   2
     1    X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2               OUTCOM   3
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY   2
C                                                                       PLOTT1   7
C    IF THERE IS ONLY ONE POINT TO BE PLOTTED, FORGET IT AND PLOT FOUR  PLOTT1   8
C    POINTS ON THE SCREEN TO FORM A LARGE X COVERING THE SCREEN        PLOTT1   9
      IF(NN1.GT.1) GO TO 5                                              PLOTT1  10
      NN1=4                                                             PLOTT1  11
      X1(1) = X1MIN                                                     PLOTT1  12
      X1(2) = X1MAX                                                     PLOTT1  13
      X1(3) = X1MIN                                                     PLOTT1  14
      X1(4) = X1MAX                                                     PLOTT1  15
      Y1(1) = Y1MIN                                                     PLOTT1  16
      Y1(2) = Y1MAX                                                     PLOTT1  17
      Y1(3) = Y1MAX                                                     PLOTT1  18
      Y1(4) = Y1MIN                                                     PLOTT1  19
C                                                                       PLOTT1  20
C    FIND THE LARGEST AND SMALLEST VALUES OF Y1                        PLOTT1  21
    5 CALL AMXMN1(YMAX,YMIN)                                            PLOTT1  22
C                                                                       PLOTT1  23
C    CREATE THE POLYLINE ENTITY FOR NXY1                               PLOTT1  24
      NGRAF = 0                                                         PLOTT1  25
      NXY1(5) = 0                                                       PLOTT1  26
      CALL DLETE(NXY1)                                                  PLOTT1  27
      XMIN = X1(1)                                                      PLOTT1  28
      XMAX = X1(NN1)                                                    PLOTT1  29
      IF(X1MIN.LT.XMIN) XMIN = X1MIN                                    PLOTT1  30
      IF(X1MAX.GT.XMAX) XMAX = X1MAX                                    PLOTT1  31
      IF(Y1MIN.LT.YMIN) YMIN = Y1MIN                                    PLOTT1  32
      IF(Y1MAX.GT.YMAX) YMAX = Y1MAX                                    PLOTT1  33
      IF(NN1.GT.60) NGRAF = 1                                           PLOTT1  34
      IF(NN1.GT.120) NGRAF = 2                                          PLOTT1  35
      IF(NGRAF.EQ.0) GO TO 30                                           PLOTT1  36
      DO 20 I=1,NGRAF                                                   PLOTT1  37
      NXY1(5) = I                                                       PLOTT1  38
      CALL PLYLN(NXY1,1,X1(60*I-59),Y1(60*I-59),60)                     PLOTT1  39
   20 CONTINUE                                                          PLOTT1  40
   30 NXY1(5) = 60                                                      PLOTT1  41
      IF((NN1-60*NGRAF-1).LE.0) GO TO 40                                PLOTT1  42
      CALL PLYLN(NXY1,1,X1(1+60*NGRAF),Y1(1+60*NGRAF),NN1-60*NGRAF-1)   PLOTT1  43
   40 CALL AREA1(XMIN,XMAX,YMIN,YMAX)                                   PLOTT1  44
      NXY1(5) =0                                                        PLOTT1  45
C                                                                       PLOTT1  46
C    DISPLAY THE POLYLINE ENTITY FOR NXY1                              PLOTT1  47
      CALL GENDF(NXY1,0)                                                PLOTT1  48
      RETURN                                                            PLOTT1  49
      END                                                               PLOTT1  50
```

```
      SUBROUTINE PLOTT2(X1MIN,X1MAX,Y1MIN,Y1MAX,Y2MIN,Y2MAX,J)          PLOTT2   2
C                                                                       PLOTT2   3
C   THIS SUBROUTINE DISPLAYS TWO CURVES IN THE TWO SUBAREAS OF THE      PLOTT2   4
C   GRAPHIC DISPLAY AREA                                                PLOTT2   5
C                                                                       PLOTT2   6
      COMMON/OUTCOM/                                                    OUTCOM   2
     1    X1(160)   ,Y1(160)   ,Y2(160)   ,NN1       ,NN2               OUTCOM   3
      COMMON/COMNXY/NXY1(6)   ,NXY2(6)                                  COMNXY   2
C                                                                       PLOTT2   9
C   IF THERE IS ONLY ONE POINT TO BE PLOTTED, FORGET IT AND PLOT FOUR   PLOTT2  10
C   POINTS ON THE SCREEN TO FORM A LARGE X COVERING THE SCREEN          PLOTT2  11
      IF(NN1.GT.1) GO TO 5                                              PLOTT2  12
      NN1=4                                                             PLOTT2  13
      X1(1) = X1MIN                                                     PLOTT2  14
      X1(2) = X1MAX                                                     PLOTT2  15
      X1(3) = X1MIN                                                     PLOTT2  16
      X1(4) = X1MAX                                                     PLOTT2  17
      Y1(1) = Y1MIN                                                     PLOTT2  18
      Y1(2) = Y1MAX                                                     PLOTT2  19
      Y1(3) = Y1MAX                                                     PLOTT2  20
      Y1(4) = Y1MIN                                                     PLOTT2  21
      Y2(1) = Y2MIN                                                     PLOTT2  22
      Y2(2) = Y2MAX                                                     PLOTT2  23
      Y2(3) = Y2MAX                                                     PLOTT2  24
      Y2(4) = Y2MIN                                                     PLOTT2  25
C                                                                       PLOTT2  26
C   FIND THE LARGEST AND SMALLEST VALUES OF Y1                          PLOTT2  27
    5 CALL AMXMN1(YMAX,YMIN)                                            PLOTT2  28
C                                                                       PLOTT2  29
C   CREATE THE POLYLINE ENTITY FOR NXY1                                 PLOTT2  30
      NGRAF = 0                                                         PLOTT2  31
      NXY1(5) = 0                                                       PLOTT2  32
      CALL DLETE(NXY1)                                                  PLOTT2  33
      XMIN = X1(1)                                                      PLOTT2  34
      XMAX = X1(NN1)                                                    PLOTT2  35
      IF(X1MIN.LT.XMIN) XMIN = X1MIN                                    PLOTT2  36
      IF(X1MAX.GT.XMAX) XMAX = X1MAX                                    PLOTT2  37
      IF(Y1MIN.LT.YMIN) YMIN = Y1MIN                                    PLOTT2  38
      IF(Y1MAX.GT.YMAX) YMAX = Y1MAX                                    PLOTT2  39
      IF(NN1.GT.60) NGRAF = 1                                           PLOTT2  40
      IF(NN1.GT.120) NGRAF = 2                                          PLOTT2  41
      IF(NGRAF.EQ.0) GO TO 30                                           PLOTT2  42
      DO 20 I=1,NGRAF                                                   PLOTT2  43
      NXY1(5) = I                                                       PLOTT2  44
      CALL PLYLN(NXY1,1,X1(60*I-59),Y1(60*I-59),60)                     PLOTT2  45
   20 CONTINUE                                                          PLOTT2  46
   30 NXY1(5) = 60                                                      PLOTT2  47
      IF((NN1-60*NGRAF-1).LE.0) GO TO 32                                PLOTT2  48
      CALL PLYLN(NXY1,1,X1(1+60*NGRAF),Y1(1+60*NGRAF),NN1-60*NGRAF-1)   PLOTT2  49
   32 CALL AREA2(XMIN,XMAX,YMIN,YMAX,1)                                 PLOTT2  50
      NXY1(5) = 0                                                       PLOTT2  51
C                                                                       PLOTT2  52
C   DISPLAY THE POLYLINE ENTITY FOR NXY1                                PLOTT2  53
      CALL GENDF(NXY1,0)                                                PLOTT2  54
      IF(J.EQ.1) RETURN                                                 PLOTT2  55
      CALL AMXMN2(YMAX,YMIN)                                            PLOTT2  56
      NXY2(5) = 0                                                       PLOTT2  57
      CALL DLETE(NXY2)                                                  PLOTT2  58
      IF(NN1.LE.1) GO TO 50                                             PLOTT2  59
      IF(Y2MIN.LT.YMIN) YMIN = Y2MIN                                    PLOTT2  60
      IF(Y2MAX.GT.YMAX) YMAX = Y2MAX                                    PLOTT2  61
C                                                                       PLOTT2  62
C   CREATE THE POLYLINE ENTITY FOR NXY2                                 PLOTT2  63
      IF(NGRAF.EQ.0) GO TO 50                                           PLOTT2  64
      DO 40 I=1,NGRAF                                                   PLOTT2  65
      NXY2(5) = I                                                       PLOTT2  66
   40 CALL PLYLN(NXY2,1,X1(60*I-59),Y2(60*I-59),60)                     PLOTT2  67
   50 NXY2(5) = 61                                                      PLOTT2  68
      IF((NN1-60*NGRAF-1).LE.0) GO TO 60                                PLOTT2  69
```

```
      CALL PLYLN(NXY2,1,X1(1+60*NGRAF),Y2(1+60*NGRAF),NN1-60*NGRAF-1)     PLOTT2
   60 CALL AREA2(XMIN,XMAX,YMIN,YMAX,2)                                   PLOTT2
      NXY2(5) = 0                                                         PLOTT2
C                                                                         PLOTT2
C    DISPLAY THE POLYLINE ENTITY FOR NXY2                                 PLOTT2
      CALL GENDF(NXY2,0)                                                  PLOTT2
      RETURN                                                              PLOTT2
      END                                                                 PLOTT2
```

```
      SUBROUTINE AREA1(XMIN,XMAX,YMIN,YMAX)                            AREA1     2
C                                                                      AREA1     3
C   THIS SUBROUTINE DETERMINES THE GRID DISPLAY FOR A GRAPH COVERING   AREA1     4
C   THE ENTIRE GRAPHIC DISPLAY AREA                                    AREA1     5
C                                                                      AREA1     6
      COMMON/ISSCAL/IDSCAL                                             ISSCAL    2
      DIMENSION ALIM(4),USER(4)                                        AREA1     8
      DATA ALIM/-40.,-40.,57.,57./                                     AREA1     9
      DX = XMAX-XMIN                                                   AREA1    10
      DY = YMAX-YMIN                                                   AREA1    11
      USER(1) = XMIN                                                   AREA1    12
      USER(2) = YMIN                                                   AREA1    13
      USER(3) = XMAX                                                   AREA1    14
      USER(4) = YMAX                                                   AREA1    15
      IDSCAL = 2                                                       AREA1    16
      CALL SSCAL(IDSCAL,ALIM,USER)                                     AREA1    17
      CALL ASCAL(IDSCAL)                                               AREA1    18
      CALL GRDNM(IDSCAL)                                               AREA1    19
      CALL CGRID1V(2,XMIN,XMAX,     YMIN,YMAX,DX,DY,0,0,1,1,6, 6)      AREA1    20
      CALL RTNID(IDA)                                                  AREA1    21
      RETURN                                                           AREA1    22
      END                                                              AREA1    23
```

```
      SUBROUTINE AREA2(XMIN,XMAX,YMIN,YMAX,ID)                          AREA2      2
C                                                                       AREA2      3
C     THIS SUBROUTINE CREATES THE GRID DISPLAY FOR A GRAPH COVERING A   AREA2      4
C     SUBAREA OF THE ENTIRE GRAPHIC DISPLAY AREA DEPENDING ON ID        AREA2      5
C                                                                       AREA2      6
      COMMON/ISSCAL/IDSCAL                                              ISSCAL     2
      DIMENSION ALIM(4,2),USER(4,2)                                     AREA2      8
      DATA ALIM/-40.,-40.,57.,10.,-40.,17.,57.,57./                     AREA2      9
      USER(1,ID) = XMIN                                                 AREA2     10
      USER(2,ID) = YMIN                                                 AREA2     11
      USER(3,ID) = XMAX                                                 AREA2     12
      USER(4,ID) = YMAX                                                 AREA2     13
      DX = XMAX-XMIN                                                    AREA2     14
      DY = YMAX-YMIN                                                    AREA2     15
      IDSCAL = ID+2                                                     AREA2     16
      CALL SSCAL(IDSCAL,ALIM(1,ID),USER(1,ID))                          AREA2     17
      CALL ASCAL(IDSCAL)                                                AREA2     18
      CALL GRDNM(IDSCAL)                                                AREA2     19
      CALL CGRIDIV(2,XMIN,XMAX,YMIN,YMAX,DX,DY,0,0,1,1,6,6)             AREA2     20
      CALL RTNID(IDA)                                                   AREA2     21
      RETURN                                                            AREA2     22
      END                                                               AREA2     23
```

```
      SUBROUTINE AMXMN1(Y1MAX,Y1MIN)                                    AMXMN1     2
C                                                                       AMXMN1     3
C   THIS SUBROUTINE DETERMINES THE LARGEST AND SMALLEST VALUES FOR Y1   AMXMN1     4
C                                                                       AMXMN1     5
      COMMON/OUTCOM/                                                    OUTCOM     2
     1    X1(160)  ,Y1(160)  ,Y2(160)  ,NN1       ,NN2                  OUTCOM     3
      Y1MAX = Y1(1)                                                     AMXMN1     7
      Y1MIN = Y1(1)                                                     AMXMN1     8
      DO 20 I=2,NN1                                                     AMXMN1     9
      IF(Y1(I)-Y1MAX) 15,15,12                                          AMXMN1    10
   12 Y1MAX = Y1(I)                                                     AMXMN1    11
      GO TO 20                                                          AMXMN1    12
   15 IF(Y1(I)-Y1MIN)18,20,20                                           AMXMN1    13
   18 Y1MIN = Y1(I)                                                     AMXMN1    14
   20 CONTINUE                                                          AMXMN1    15
      RETURN                                                            AMXMN1    16
      END                                                               AMXMN1    17
```

```
      SUBROUTINE AMXMN2(Y2MAX,Y2MIN)                                    AMXMN2   2
C                                                                       AMXMN2   3
C   THIS SUBROUTINE DETERMINES THE LARGEST AND SMALLEST VALUES FOR Y2   AMXMN2   4
C                                                                       AMXMN2   5
      COMMON/OUTCOM/                                                    OUTCOM   2
     1     X1(160)    ,Y1(160)    ,Y2(160)    ,NN1        ,NN2          OUTCOM   3
      Y2MAX = Y2(1)                                                     AMXMN2   7
      Y2MIN = Y2(1)                                                     AMXMN2   8
      DO 30 I=2,NN1                                                     AMXMN2   9
      IF(Y2(I) -Y2MAX) 25,25,22                                         AMXMN2  10
   22 Y2MAX = Y2(I)                                                     AMXMN2  11
      GO TO 30                                                          AMXMN2  12
   25 IF(Y2(I) -Y2MIN) 28,30,30                                         AMXMN2  13
   28 Y2MIN = Y2(I)                                                     AMXMN2  14
   30 CONTINUE                                                          AMXMN2  15
      RETURN                                                            AMXMN2  16
      END                                                               AMXMN2  17
```

REFERENCES

1. Tai, T. C., "Application of the Method of Integral Relations to Transonic Airfoil Problems: Part I — Inviscid Supercritical Flow over Symmetrical Airfoil at Zero Angle of Attack," NSRDC Report 3424 (Sep 1970); also presented as Paper 71-98, AIAA 9th Aerospace Sciences Meeting, New York, N. Y. (Jan 1971).

2. Tai, T. C., "Application of the Method of Integral Relations to Transonic Airfoil Problems: Part II — Inviscid Supercritical Flow About Lifting Airfoils with Embedded Shock Wave," NSRDC Report 3424 (Jul 1972); also presented as Paper 73-658, AIAA 6th Fluid and Plasma Dynamics Conference, Palm Springs, California (Jul 1973).

3. Graham, D. J. et al., "A Systematic Investigation of Pressure Distribution at High Speeds over Five Representative NACA Low-Drag and Conventional Airfoil Sections," NACA Report 832 (1945).

4. Ahlberg, J. H. et al., "The Theory of Splines and Their Applications," Academic Press, New York (1967).

INITIAL DISTRIBUTION

Copies	
1	ARMY CHIEF OF RES & DEV
1	ABRL/TB MESZAROS
1	AMC REDSTONE SIC/DUERR
1	AMERDC
3	ONR
	1 ONR 430
	1 ONR 432/L. BRAM
	1 ONR 461
1	USNA LIB
1	NAVPGSCOL LIB
10	AIR
	1 AIR 03A
	1 AIR 03C
	1 AIR 310
	3 AIR 320
	1 AIR 50174B
	1 AIR 5108
	2 AIR 5301
1	ORD LIB
1	NAVAIRDEVCEN ADT LIB
1	NAVWPNSCEN 753 LIB
1	NOL 730 LIB
1	NWL LIB
1	NAVMISCEN
1	AFFDL/APPL MECH LIB
1	AFOSR
1	AIR FORCE AERO RES LABS
1	AFIT
1	AEDC
4	NASA
	1 HQS NASA
	1 NASA AMES RES CEN
	1 NASA LANGLEY RES CEN
	1 NASA RES CEN
1	NSF

CENTER DISTRIBUTION

1	C	0000	NELSON PERRY W
20	C	1606	TAI TSZE C
10	C	1830	ERNST HERBERT M
30	C	5614	REPORTS DISTRIBUTION
1	C	5641	LIBRARY
1	A	5642	LIBRARY
2	C	5643	AVIATION COLLECTION

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Aviation and Surface Effects Department Naval Ship Research and Development Center Bethesda, Maryland 20034	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

A COMPUTER PROGRAM THAT USES INTERACTIVE GRAPHICS TO SOLVE INVISCID TRANSONIC FLOWS OVER LIFTING AIRFOILS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Research and Development Report

5. AUTHOR(S) *(First name, middle initial, last name)*

Douglas A. Lien and Tsze C. Tai

6. REPORT DATE September 1973	7a. TOTAL NO. OF PAGES 201	7b. NO. OF REFS 4
8a. CONTRACT OR GRANT NO. NSRDC AIRTASK R230.201 b. PROJECT NO. c. NSRDC 1-1670-277 d.	9a. ORIGINATOR'S REPORT NUMBER(S) NSRDC Report 4252 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* Aero Report 1194	

10. DISTRIBUTION STATEMENT

FOR OFFICIAL USE ONLY

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Commander Naval Air Systems Command (AIR-320) Washington, D. C. 20360

13. ABSTRACT

A computer program that utilizes the method of integral relations has been developed at the Naval Ship Research and Development Center for use in determining the inviscid transonic flows past lifting airfoils. It allows for a change of entropy across the shock wave and accounts for the presence of an oblique or normal shock at the shock foot. Since many iterations of the trial and error type are required to obtain the converged flow solution, the program has been adapted for use on the interactive graphic systems of the CDC 6700 computer. This minimizes the man-machine interaction time involved with such iterations. It has been applied to several airfoil cases with supercritical flow on the upper surface and subcritical flow on the lower surface. The theoretical basis for this program has previously been reported. This report documents the computer program which is written in the language of FORTRAN Extended Version 3.0.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Transonic Flow						
Interactive Computer Graphics						
Lifting Airfoil						
Shock Wave						
Method of Integral Relations						